



**Titre:** Construction de règles de Golomb optimales  
Title:

**Auteur:** Sien Cuong Hoa  
Author:

**Date:** 1999

**Type:** Mémoire ou thèse / Dissertation or Thesis

**Référence:** Hoa, S. C. (1999). Construction de règles de Golomb optimales [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie.  
Citation: <https://publications.polymtl.ca/8596/>

 **Document en libre accès dans PolyPublie**  
Open Access document in PolyPublie

**URL de PolyPublie:** <https://publications.polymtl.ca/8596/>  
PolyPublie URL:

**Directeurs de  
recherche:**  
Advisors:

**Programme:** Non spécifié  
Program:

UNIVERSITÉ DE MONTRÉAL

Construction de règles de Golomb optimales

SIEN CUONG HOA

DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL  
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION  
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES  
(MATHÉMATIQUES APPLIQUÉES)

DECEMBRE 1999



National Library  
of Canada

Acquisitions and  
Bibliographic Services

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque nationale  
du Canada

Acquisitions et  
services bibliographiques

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file Votre référence*

*Our file Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-48856-X

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

## Construction de règles de Golomb optimales

présenté par: HOA, Sien Cuong

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. HACCOUN David, Ph.D, président

Mme JAUMARD Brigitte, T. Doct., T. Hab., membre et directrice de recherche

Mme JANSSEN Jeannette, Ph.D, membre

## REMERCIEMENTS

Je désire exprimer ma gratitude à madame Brigitte Jaumard pour le sujet de recherche sur lequel elle m'a suggéré de travailler. Je lui exprime aussi ma reconnaissance pour le temps qu'elle m'a consacré, pour les directives et les corrections qu'elle m'a proposées dans la rédaction de ce mémoire. Je la remercie également pour le soutien financier qu'elle m'a accordé.

Je remercie également Monsieur David Haccoun et Madame Jeannette Janssen pour avoir accepté de faire partie du jury.

Je tiens également à remercier Denis Boyer pour les problèmes tests qu'il a générés pour moi et qui m'ont permis de tester le programme d'affectation de fréquences. Je remercie également tous ceux et celles qui ont lu attentivement ce mémoire pour faire part de leurs commentaires.

## RÉSUMÉ

Nous nous intéressons à la construction de règles de Golomb optimales et nous discutons de deux de leurs applications en télécommunication, soit de l'élimination des contraintes d'intermodulation dans les réseaux de la téléphonie cellulaire, et de l'optimisation des codes convolutionnels. Les règles de Golomb sont des règles avec des marques espacées de telle sorte que les différences entre deux marques quelconques soient toutes distinctes. Une règle de Golomb est optimale si elle est de longueur minimale pour un nombre de marques fixé.

Plusieurs auteurs se sont intéressés à la construction de règles de Golomb optimales. Tous se sentent heurtés à la difficulté de trouver des algorithmes efficaces, et aujourd'hui encore trouver une règle optimale de 21 marques requiert environ 65 ans sur un pentium II avec 266 MHz. Le projet de recherche de ce mémoire concerne l'amélioration de GARSP qui est l'algorithme le plus efficace aujourd'hui pour calculer une règle de Golomb optimale. Le travail réalisé a été orienté vers l'amélioration des tests de réduction et sur l'introduction de bornes inférieures plus précises. Le résultat obtenu est un algorithme qui améliore les temps de calcul de GARSP d'environ 40%.

**Mots clés :** règle de Golomb, bitmap, intermodulation, affectation de fréquences, différences de triangles, réseau de la téléphonie cellulaire, graphes semi-gracieux.

## ABSTRACT

We are interested in designing optimal Golomb rulers and we discuss two of their applications in telecommunication, i.e., elimination of intermodulation constraints in cellular networks, and optimization of convolutional codes. Golomb rulers are rulers with marks at position such that differences between any two marks are all distinct. A Golomb ruler is optimal if it has a minimal length for a fixed number of marks.

Several authors have been interested in designing optimal Golomb rulers. However, finding an efficient algorithm is not an easy task. Still today, finding an optimal 21 mark rulers requires more than 65 years on a 266 MHz Pentium II. The topic of this master thesis consists in the improvement of GARSP, one of the most efficient algorithms currently known today. This thesis will be centered on improving reduction tests and on the introduction of more precise lower bounds. The result is an algorithm which improves the computing time by 40%.

**Key words :** Optimum Golomb ruler, bitmap, intermodulation, frequency assignment, difference triangle sets, cellular network, semi-graceful graphs.

## TABLE DES MATIÈRES

<b>REMERCIEMENTS</b> . . . . .	<b>iv</b>
<b>RÉSUMÉ</b> . . . . .	<b>v</b>
<b>ABSTRACT</b> . . . . .	<b>vi</b>
<b>TABLE DES MATIÈRES</b> . . . . .	<b>vii</b>
<b>LISTE DES TABLEAUX</b> . . . . .	<b>x</b>
<b>LISTE DES FIGURES</b> . . . . .	<b>xi</b>
<b>LISTE DES ANNEXES</b> . . . . .	<b>xii</b>
<b>CHAPITRE 1 : REVUE DE LA LITTÉRATURE</b> . . . . .	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Définitions . . . . .	3
1.2.1 Règles de Golomb . . . . .	3
1.2.2 Triangles de différences . . . . .	5
1.2.3 Numérotation d'un graphe . . . . .	6
1.2.4 Ensemble de différences cycliques . . . . .	7
1.3 Historique des règles de Golomb optimales . . . . .	9
1.4 Règles de Golomb optimales connues . . . . .	10
1.5 Une application importante des règles de Golomb : les codes convo- lutionnels systématiques . . . . .	12
1.5.1 Introduction . . . . .	12
1.5.2 Codeur convolutionnel systématique . . . . .	13
1.5.3 Décodeur à seuil . . . . .	15



1.6	Algorithme de recherche exacte : GARSP . . . . .	16
1.6.1	Définitions et structures bitmaps . . . . .	16
1.6.2	GARSP . . . . .	21
1.7	Algorithme de recherche heuristique : plans projectifs et semi-affines	26
1.7.1	Définitions . . . . .	26
1.7.2	Construction des ensembles de différences cycliques . . . . .	27

## CHAPITRE 2 : VERSIONS AMÉLIORÉES DE L'ALGORITHME

<b>GARSP</b>	<b>33</b>
2.1 Introduction	33
2.2 G-DFS1	33
2.3 Stratégie de recherche - DFS	34
2.4 G-DFS2	38
2.5 Tests de réduction	42
2.5.1 Élimination des règles symétriques	42
2.5.2 Test de la position limite	42
2.5.3 Test de la borne inférieure	43
2.5.3.1 Définition de la borne inférieure	43
2.5.3.2 Calcul de la borne inférieure	45
2.5.3.3 Borne inférieure dans G-DFS1	45
2.5.3.4 Borne inférieure dans G-DFS2	52
2.6 Résultats et analyse	54
2.7 Conclusion	61

## CHAPITRE 3 : INTERMODULATION ET

<b>AFFECTATION DE FRÉQUENCES . . . . .</b>	<b>63</b>
3.1 Introduction . . . . .	63
3.2 Définition . . . . .	65

3.3	Affectation de fréquences avec élimination de l'intermodulation du 3 <sup>e</sup>	
	ordre . . . . .	68
3.3.1	Interférences co-canal . . . . .	68
3.3.2	Interférences de canaux adjacents . . . . .	69
3.3.3	Contraintes d'espacement sur une antenne . . . . .	69
3.3.4	Intermodulation du 3 <sup>e</sup> ordre . . . . .	70
	3.3.4.1 Deux canaux . . . . .	70
	3.3.4.2 Trois canaux . . . . .	70
3.4	Règle de Golomb et problème d'intermodulation . . . . .	73
3.5	Algorithme de recherche . . . . .	78
3.6	Borne inférieure . . . . .	81
3.7	Résultats . . . . .	83
3.8	Conclusion . . . . .	86
<b>BIBLIOGRAPHIE . . . . .</b>		<b>87</b>
<b>ANNEXE A . . . . .</b>		<b>93</b>
<b>ANNEXE B . . . . .</b>		<b>98</b>

## LISTE DES TABLEAUX

1.1	<i>Règles de Golomb optimales</i>	10
1.2	<i>Exemple d'un codage</i>	15
2.1	<i>Temps de calcul des règles de Golomb</i>	54
2.2	<i>Nombre de noeuds examinés</i>	55
2.3	<i>Bornes supérieures</i>	57
2.4	<i>Bornes supérieures pour un nombre premier de marques</i>	59
3.1	<i>Affectation de fréquences</i>	85

## LISTE DES FIGURES

1.1	<i>Une règle de Golomb avec 5 marques . . . . .</i>	3
1.2	<i>Une règle de Golomb parfaite avec 4 marques . . . . .</i>	4
1.3	<i>Deux règles de Golomb symétriques . . . . .</i>	4
1.4	<i>Triangles de différences correspondant à trois règles de Golomb disjointes . . . . .</i>	5
1.5	<i>Numérotation d'un graphe complet de 5 sommets . . . . .</i>	6
1.6	<i>Table des différences . . . . .</i>	8
1.7	<i>Codeur convolutionnel systématique . . . . .</i>	14
1.8	<i>État du Codeur . . . . .</i>	15
1.9	<i>Positions de marques . . . . .</i>	21
2.1	<i>Numérotation des différences des marques consécutives . . . . .</i>	35
2.2	<i>Distances interdites . . . . .</i>	36
3.1	<i>Site . . . . .</i>	65
3.2	<i>Interférences co-canal . . . . .</i>	68
3.3	<i>Interférences de canaux adjacents . . . . .</i>	69
3.4	<i>Intermodulation du 3<sup>e</sup> ordre . . . . .</i>	72
3.5	<i>Réseau à 2 sites . . . . .</i>	73
3.6	<i>Sous-réseau à 20 cellules . . . . .</i>	84

## LISTE DES ANNEXES

ANNEXE A : Le problème d'affectation . . . . .	93
A.1 Cellules d'un site . . . . .	94
A.2 Demande en canaux de chaque cellule . . . . .	95
A.3 Matrice $M^{sep}$ . . . . .	96
A.4 Matrice $M^{mod}$ . . . . .	97
ANNEXE B : Les contributions . . . . .	98
B.1 Les contribution . . . . .	99
B.1.1 Les règles de Golomb . . . . .	99
B.1.2 Le problème d'affectation de fréquences . . . . .	99

# CHAPITRE 1

## REVUE DE LA LITTÉRATURE

### 1.1 Introduction

Les règles de Golomb sont des règles avec des marques espacées inégalement de telle sorte que les distances mesurées entre deux marques quelconques sont distinctes. Une règle est dite optimale si sa longueur est minimale pour un nombre fixé de marques. D'abord étudiées comme une curiosité mathématique (Dewdney [7] et [8]), les règles de Golomb sont aujourd'hui étudiées par plusieurs mathématiciens et chercheurs. Elles interviennent dans de nombreuses applications où des contraintes d'espacement doivent être respectées, par exemple en cristallographie [4], en radio astronomie [7], en télécommunication [10], en théorie du codage [5] [17]. Le lecteur intéressé peut consulter les deux articles de vulgarisation de Dewdney [7] et [8] parus dans le Scientific American pour les références de base.

Le professeur Solomon W. Golomb, qui a donné son nom aux règles, a été l'un des premiers chercheurs à analyser les règles de Golomb. Après une percée importante de Robinson [25] en 1967 qui a conduit à plusieurs règles de Golomb presque optimales ou optimales de 3 à 24 marques, plusieurs algorithmes ont été proposés pour calculer ces règles. Ces algorithmes reposent principalement sur deux types de méthodes. D'une part, les méthodes exactes qui utilisent des techniques d'énumération implicite. D'autre part, les méthodes heuristiques qui font appel à des plans projectifs ou semi-affines. Plus récemment, d'autres méthodes heuristiques du type algorithmes génétiques ont été proposées par Soliday, Homaifar et Lebby [27].

Dans le cadre de ce mémoire, nous nous intéressons aux méthodes exactes. Notre point de départ est l'algorithme GARSP dû à Rakin et Garry [32] et qui

constitue l'algorithme le plus efficace aujourd'hui. Nous étudions comment insérer dans GARSP de nouveaux tests de réduction et de nouvelles bornes inférieures récemment proposées par Hansen, Jaumard et Meyer [16]. La difficulté réside dans la détermination du meilleur équilibre temps de calcul - efficacité des bornes sachant que plus les bornes sont précises, plus elles sont coûteuses à calculer.

En second lieu, nous aborderons le problème d'affectation de fréquences dans le cadre d'un réseau de téléphonie cellulaire sous contraintes d'intermodulation et d'interférences co-canal et de canaux adjacents. Nous proposons de développer un algorithme permettant de vérifier s'il existe une affectation de fréquences réalisable, c'est-à-dire satisfaisant toutes les contraintes d'interférences pour de petits réseaux de la téléphonie cellulaire.

Ce mémoire sera donc divisé en trois chapitres : les deux premiers porteront sur les règles de Golomb et le dernier sur le problème d'affectation de fréquences.

## 1.2 Définitions

### 1.2.1 Règle de Golomb

Une règle de Golomb est définie comme un ensemble de divisions sur une échelle linéaire telles que les distances mesurées entre toutes les paires de divisions sont distinctes. Chacune des divisions correspond à une **marque** et l'ensemble des distances entre les paires de marques sont les **différences** mesurées par la règle. Par convention, la première marque est fixée à zéro. On notera par  $M$  le nombre de marques.

Nous représentons ci-dessous à la figure 1.1 une règle de Golomb avec 5 marques, ainsi que l'ensemble des différences entre deux marques : tout d'abord les positions des marques qui définissent les différences par rapport à la première marque (position 0), ensuite les différences entre deux marques consécutives et finalement les différences entre deux marques non consécutives d'abord espacées d'une marque, puis deux et ainsi de suite.

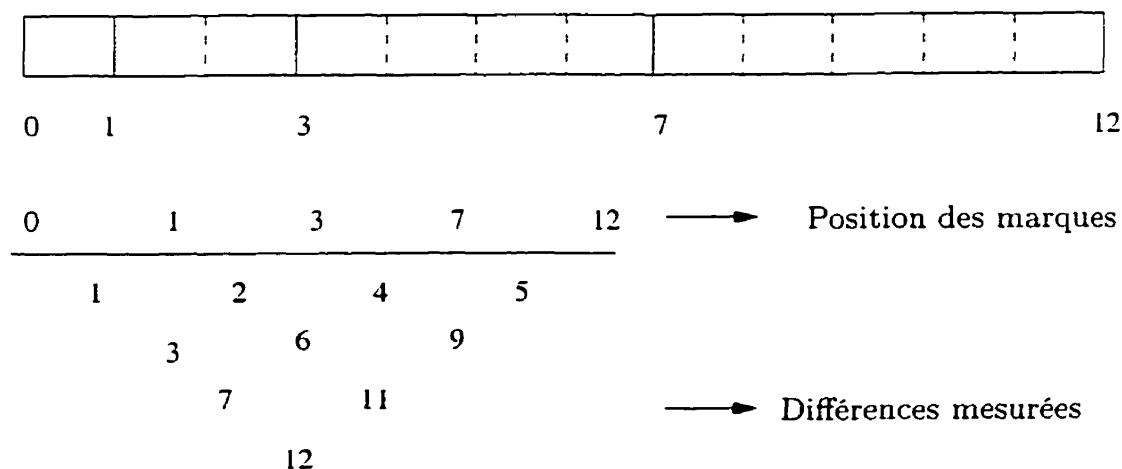


Figure 1.1 – Une règle de Golomb avec 5 marques

La **longueur d'une règle** est la différence mesurée entre la première et la



dernière marque. Une règle est dite **optimale** si sa longueur est minimale pour un nombre fixe de marques. Par conséquent la règle avec 5 marques et de longueur 12 représentée à la figure 1.1 est non optimale, mais les deux de la figure 1.3 de longueur 11 le sont. Une règle de Golomb est dite **parfaite** si elle mesure toutes les différences entre 1 et  $\frac{M(M-1)}{2}$ . Les règles de Golomb des figures 1.1 et 1.3 ne sont pas parfaites, puisque les valeurs 8 et 10 sont manquantes pour la règle de la figure 1.1 et la valeur 6 est manquante pour les deux règles de la figure 1.3. À la figure 1.2, nous avons représenté une règle parfaite avec 4 marques. On peut montrer qu'il n'existe pas de règle parfaite avec plus de 4 marques, voir Golomb [13].

0	1	4	6
	1	3	2
	4	5	
		6	

Figure 1.2 – Une règle de Golomb parfaite avec 4 marques.

Pour toute règle de Golomb, il existe une deuxième règle de même longueur qui est l'image miroir de la première. Ces règles sont dites **règles symétriques**. Deux règles sont symétriques si elles mesurent le même ensemble de différences, mais sont telles que les différences entre les paires de marques consécutives d'une des règles sont placées dans l'ordre inverse de l'autre (voir figure 1.3 ci-dessous).

0	1	4	9	11
	1	3	5	2
		4	8	7
			9	10
				11

0	2	7	10	11
	2	5	3	1
		7	8	4
			10	9
				11

Figure 1.3 – Deux règles de Golomb symétriques.

On retrouve dans la littérature des problèmes très proches des règles de Golomb. Il s'agit des **triangles de différences**, **ensembles disjoints de différences**, des

problèmes de **numérotation d'un graphe** et celui des **ensembles de différences cycliques**. Nous rappelons la définition de ces problèmes dans les sous-sections suivantes et nous identifions leurs liens avec les règles de Golomb.

### 1.2.2 Triangles de différences

Le terme triangle de différences (Kløve [19],[20]) tire son nom du fait que les différences mesurées par une règle de Golomb peuvent être décrites par un triangle dans lequel la première ligne est utilisée pour noter les positions des marques, la deuxième pour les différences mesurées entre deux marques consécutives, la troisième pour les différences entre les paires de marques espacées d'une marque, puis de deux marques et ainsi de suite. Généralement dans le cas des triangles de différences, les auteurs s'intéressent non pas à une seule règle, mais à un ensemble disjoint de règles ayant un même nombre de marques.

Les triangles de différences sont ainsi référencés sous l'acronyme  $(I, J)$ -DTS (Difference Triangle Sets). Les paramètres  $I$  et  $J$  indiquent respectivement le nombre de règles et le nombre de différences consécutives mesurées par chaque règle (soit encore  $M = J + 1$ ). Les triangles de différences sont donc une extension des règles de Golomb. L'objectif est de minimiser le maximum des longueurs minimales de chacune des règles. La figure ci-dessous représente un exemple d'un  $(3,3)$ -DTS optimal.

0	1	6	13	0	3	14	18	0	2	10	19
1	5	7		3	11	4		2	8	9	
	6	12			14	15			10	17	
		13				18				19	

Figure 1.4 – Triangles de différences correspondant à trois règles de Golomb disjointes.

### 1.2.3 Numérotation d'un graphe

Soit  $G(V, A)$  un graphe quelconque, c'est-à-dire un ensemble de sommets noté  $V$  et un ensemble d'arêtes noté  $A$  qui relie les sommets du graphe. Le lecteur désirant en connaître plus sur la théorie des graphes pourra se référer à Berge [3] pour les définitions de base. Le problème de numérotation d'un graphe (voir Bloom et Golomb [4], Gibbs et Slater [12], Golomb [13]) consiste à affecter des valeurs positives  $L(v)$  à chacun des sommets  $v$  d'un graphe de telle sorte que :

- toutes les valeurs  $L(v)$  sont distinctes,
- toutes les valeurs  $|L(v) - L(u)|$  associées à une arête  $\{u, v\} \in A$  sont distinctes,
- la valeur  $\max_{v \in V} L(v)$  est minimale.

On peut établir une correspondance avec les règles de Golomb en associant les marques aux valeurs associées aux sommets, et les différences entre les marques aux valeurs  $|L(v) - L(u)|$  associées aux arêtes.

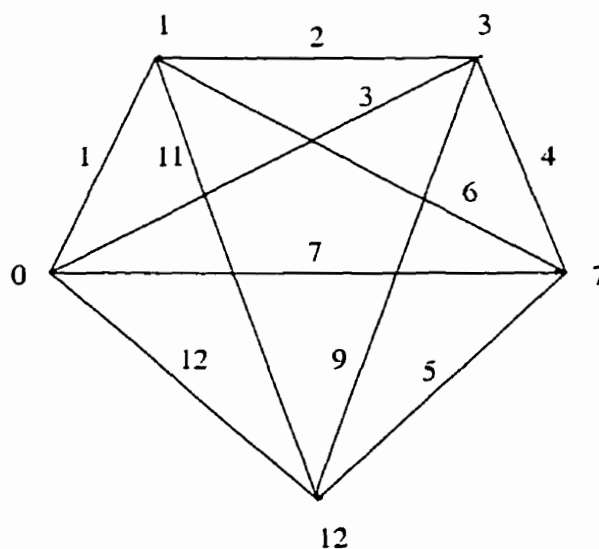


Figure 1.5 – Numérotation d'un graphe complet de 5 sommets

En d'autres termes, trouver une règle de Golomb avec  $M$  marques est équivalent à numérotter un graphe complet de  $M$  sommets, soit un graphe qui contient

toutes les arêtes possibles. Voir la figure 1.5 pour un exemple de numérotation d'un graphe complet. Il faut cependant noter que la numérotation d'un graphe est définie pour des graphes quelconques qui ne sont pas nécessairement complets, et constitue donc un problème plus général que celui de trouver des règles de Golomb optimales.

Comme pour les règles de Golomb, il existe une notion de perfection pour les problèmes de numérotation d'un graphe qui conduit aux graphes gracieux (Gardner [11]). Un graphe est dit gracieux s'il admet une numérotation gracieuse, c'est-à-dire si  $\max_{v \in V} L(v)$  est égale à la cardinalité de  $A$ , soit le nombre d'arêtes du graphe. En conséquence, les valeurs  $L(v)$  d'un graphe complet avec une numérotation gracieuse correspondent aux positions des marques d'une règle parfaite.

#### 1.2.4 Ensemble de différences cycliques

Un ensemble de différences cycliques de taille  $s$  et de module  $n = s^2 - s + 1$  est un ensemble de  $s$  entiers positifs notés  $a_1 = 0, a_2, a_3, \dots, a_s$ , tous inférieurs à  $n$ , tels que toutes les différences  $(a_i - a_j)$  modulo  $n$  pour tout  $i \neq j$  sont distinctes. Cet ensemble est noté par :

$$a_1 \quad a_2 \quad a_3 \quad \dots \quad a_s \quad (n)$$

ou  $a_1 < a_2 < a_3 \dots < a_s$ . Par exemple, l'ensemble  $S = \{0, 5, 6, 9, 19\}$  est un ensemble de différences cycliques de taille  $s = 5$  et de module  $n = 21$  et mesure l'ensemble des différences  $d_{ij}$  suivantes, où  $d_{ij} = (a_i - a_j) \text{ modulo } n$  pour tout  $i, j \in \{1, 2, \dots, s\}$  :

	0	5	6	9	19	(21)
0	0	5	6	9	19	
5	16	0	1	4	14	
6	15	20	0	3	13	
9	12	17	18	0	10	
19	2	7	8	11	0	

Figure 1.6 – Table des différences

Observer que chacune des lignes de la table des différences est une règle de Golomb de  $s$  marques. Il est donc possible de construire des règles de Golomb à partir des ensembles de différences cycliques.

### 1.3 Historique des règles de Golomb optimales

Bien que d'un énoncé très simple, le problème de trouver une règle de Golomb optimale est un problème combinatoire très difficile. Pour un nombre de marques  $M \leq 13$ , le problème peut être facilement résolu avec un algorithme d'énumération implicite de base. Shearer [26] a donné en 1990 une preuve de l'optimalité des règles de Golomb connues avec  $M = 14, 15$  et  $16$ . Puis, en 1993, Sibert a été capable de calculer les règles de Golomb optimales pour  $M = 17$  et  $18$ ; cependant ces résultats n'ont jamais été publiés (voir la page web de Shearer [33]). En 1998, Dollas, Rankin et McCracken [9] proposent un algorithme d'énumération implicite amélioré nommé SHIFT qui, implanté sur un ordinateur parallèle, permet de trouver une règle de Golomb optimale pour  $M = 19$ . Enfin, en 1999, Rankin et Garry proposent GARSP, une version encore améliorée de SHIFT, qui a permis de trouver les règles de Golomb optimales avec  $M = 20$  et  $21$  (pour  $M = 21$ , voir [31]). Un nouveau projet collectif a été mis en oeuvre sur le web (voir [31]) pour la recherche d'une règle optimale avec  $M = 22$ .

Tous les résultats obtenus pour les règles de Golomb optimales font appel à des méthodes d'énumération implicite avec un schéma d'exploration du type profondeur d'abord. Afin de réduire le nombre de sous-problèmes à explorer, divers tests de réduction et calculs de bornes inférieures sur la longueur minimale d'une règle de Golomb sont utilisés. Nous reprenons en détail ces tests et ces bornes dans la section 2.5.

## 1.4 Règles de Golomb optimales connues

Les règles de Golomb optimales sont des règles de Golomb connues avec certitude (c'est-à-dire qu'on a la preuve de leur optimalité) jusqu'à 21 marques aujourd'hui. Le tableau 1.1 ci-dessous en décrit une pour chaque valeur de  $M$  (il n'y a pas toujours unicité des règles de Golomb optimales). Le paramètre  $M$  indique le nombre de marques et la règle optimale est représentée par la position de ses marques.

Tableau 1.1 – Règles de Golomb optimales

M	Règles optimales
4	0 1 4 6
5	0 1 4 9 11
6	0 1 4 10 12 17
7	0 1 4 10 18 23 25
8	0 1 4 9 15 22 32 34
9	0 3 9 17 19 32 39 43 44
10	0 1 6 10 23 26 34 41 53 55
11	0 1 4 13 28 33 47 54 64 70 72
12	0 2 6 24 29 40 43 55 68 75 76 85
13	0 7 8 17 21 36 47 63 69 81 101 104 106
14	0 4 6 20 35 52 59 77 78 86 89 99 122 127
15	0 4 20 30 57 59 62 76 100 111 123 136 144 145 151
16	0 1 4 11 26 32 56 68 76 115 117 134 150 163 168 177
17	0 8 31 34 40 61 77 99 118 119 132 143 147 182 192 194 199
18	0 11 24 28 49 63 68 86 118 127 133 134 160 163 194 206 214 216
19	0 4 13 15 42 56 59 77 93 116 126 138 146 174 214 221 240 245 246
20	0 1 8 11 68 77 94 116 121 156 158 179 194 208 212 228 240 253 259 283
21	0 2 24 56 77 82 83 95 129 144 179 186 195 255 265 285 293 296 310 329 333

D'autres règles de Golomb sont connues pour  $M$  supérieur à 21, mais aucun chercheur n'a encore été capable d'établir leur optimalité. Le lecteur intéressé pourra

se référer à Shearer [33] ou à Miller [35] pour des règles comportant jusqu'à 211 marques.



## 1.5 Une application importante des règles de Golomb : les codes convolutionnels systématiques

### 1.5.1 Introduction

Nous avons mentionné précédemment que les règles de Golomb sont utilisées dans une variété d'applications. Dans cette section nous en présentons une en théorie du codage.

Supposons que nous ayons à transmettre de l'information via un médium tel que les lignes téléphoniques. Pour assurer que les informations transmises soient reçues correctement, un certain codage est nécessaire. En principe nous ajoutons à l'information originale un certain nombre de symboles de parité afin d'assurer une meilleure fiabilité des transmissions. En d'autres termes pour chaque bit du message envoyé, un nombre de symboles de parité associé à ce dernier est généré. Ces symboles seront envoyés avec l'information originale. Une fois le mot de code reçu, les symboles de parité permettront de vérifier si une erreur s'est glissée lors de la transmission et dans le cas échéant de corriger cette erreur.

Nous décrivons brièvement ci-dessous les codes convolutionnels (Robinson et Berstein [25], Wu [28], [29], [30]), qui sont souvent utilisés pour les transmissions sérielles à haute vitesse et plus particulièrement aux codes convolutionnels systématiques. Un code est dit systématique si son premier bit correspond au bit d'information et les autres bits aux bits de parité associés à ce dernier. L'intérêt des codes convolutionnels est qu'ils sont faciles à implanter et qu'ils ont une capacité de correction assez élevée. Dans certaines applications les symboles de parité sont générés à l'aide des codeurs construits à partir des triangles de différences ou encore

de règles de Golomb.

### 1.5.2 Codeur convolutionnel systématique

Le codeur est composé d'un ensemble de registres. Le nombre total de registres nécessaires correspond au nombre de triangles de différences utilisés. Chaque registre est composé d'un ensemble de modules de décalage placés l'un à la suite de l'autre. Le nombre total de modules dans chaque registre correspond respectivement à la longueur de la règle de Golomb qu'il représente. Au départ les modules dont la position correspond à celle d'une marque d'une règle seront initialisés à 1 (sauf pour la première marque, elle n'est pas représentée par un module de décalage) et les autres à 0. Pour chacun des registres, les modules initialisés à 1 sont reliés à un opérateur de parité (additionneur modulo-2).

À tour de rôle les bits d'informations sont décalés dans les registres, un à la fois, vers la gauche. Pour chaque bit d'information, les opérations de parité sont effectuées pour générer les  $I$  bits de parité, un pour chaque règle. Nous obtenons donc d'une part l'information originale et d'autre part un ensemble de bits de parité. Ces deux éléments forment le mot de code qui sera envoyé. Nous présentons ci-dessous un exemple d'un codeur convolutionnel systématique de mémoire  $M$  construit à partir d'un  $(2, 3)$ -DTS. Le paramètre  $M$  est la plus grande longueur mesurée par le DTS. Les deux ensembles de différences considérés sont  $\{0, 1, 10, 13\}$  et  $\{0, 4, 6, 11\}$ . L'architecture du codeur est inspirée d'un codeur proposé dans l'article de Haccoun, Lavoie et Savaria [14].

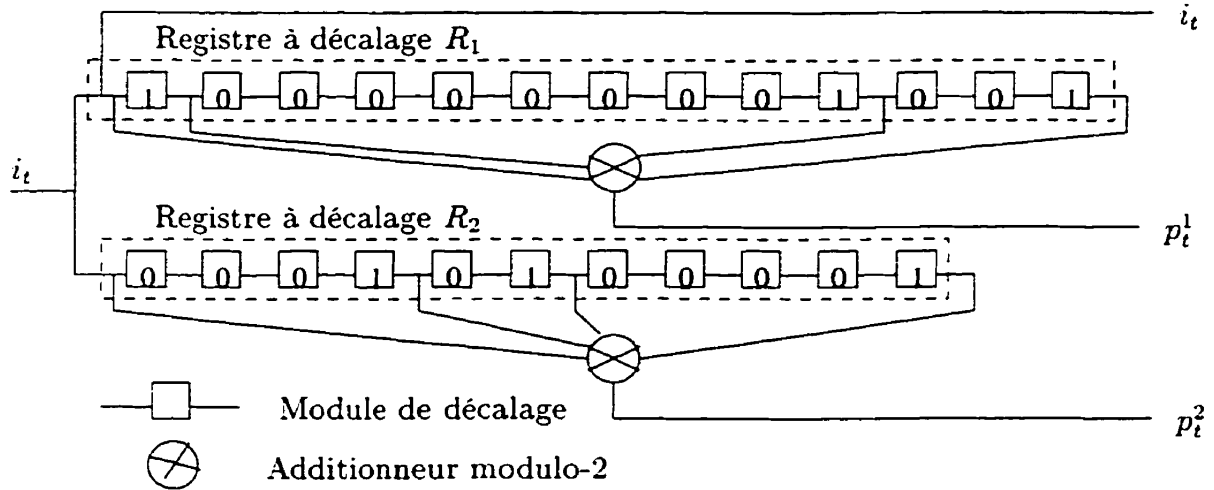


Figure 1.7 – Codeur convolutionnel systematique

Le taux de codage du codeur ci-dessous est de  $1/3$ , c'est-à-dire que pour chaque bit  $i_t$  (bit d'information au temps  $t$ ), le codeur génère un code comprenant 3 bits dont le premier est le bit d'information et les deux autres les bits de parité  $p_t^1$  et  $p_t^2$ . Nous présentons ci-dessous un exemple simple de codage des bits suivants d'information suivants (11001). À l'instant  $t = 1$ , nous avons les bits de parités suivants :

$$p_1^1 = (1 + 1 + 1 + 1) \text{ modulo } 2 = 0$$

$$p_1^2 = (1 + 1 + 1 + 1) \text{ modulo } 2 = 0.$$

À l'instant  $t = 2$ , les bits sont décalés d'une unité vers la droite de sorte que le premier bit d'information occupe maintenant le premier module de décalage. Les nouveaux bits de parité sont :

$$p_2^1 = (0 + 1 + 0 + 0) \text{ modulo } 2 = 1$$

$$p_2^2 = (0 + 0 + 0 + 0) \text{ modulo } 2 = 0.$$

Le tableau 1.2 liste les mots de code associés aux bits d'information que nous désirons transmettre et la figure 1.8 représente l'état des registres au moment du codage du dernier bit d'information.

Tableau 1.2 – Exemple d'un codage

Bit ( $i_t$ )	mot de code( $p_t^2 p_t^1 i_t$ )
1	001
0	010
0	100
1	101
1	001

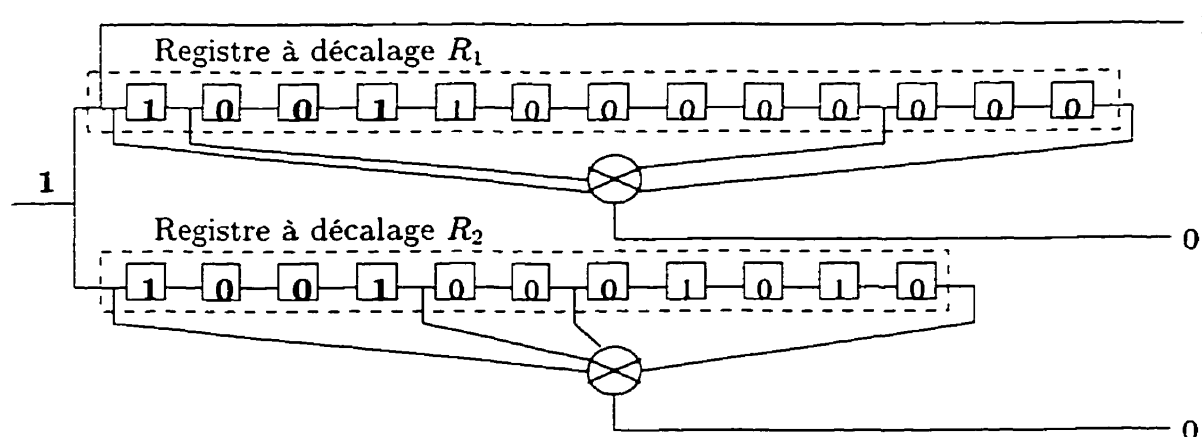


Figure 1.8 – État du codeur

### 1.5.3 Décodeur à seuil

Le décodage de l'information se fera par décodage à seuil ("Threshold decoding", voir Haccoun, Lavoie et Savaria [14][15] pour plus de détails). Le décodeur à seuil est composé d'une part d'un codeur et d'autre part des registres de syndrome qui correspondent à la représentation par des registres des règles symétriques qui ont servi à la construction du codeur. Les modules de chaque registre de syndrome correspondant aux marques de la règle symétrique sont reliés à un opérateur de logique majoritaire (voir Lavoie, Haccoun et Savaria [14] [15] pour plus de détails).

## 1.6 Algorithmes de recherche exacte : GARSP

### 1.6.1 Définitions et structures bitmaps

Nous proposons ci-dessous un algorithme de dérivation des règles optimales de Golomb. Celui-ci est une amélioration de l'algorithme **GARSP** [32] qui lui-même est une version améliorée de l'algorithme **SHIFT** [9] publié en janvier 1998. Le principe de cet algorithme est celui d'une recherche par énumération implicite qui utilise des bornes supérieures sur la longueur optimale d'une règle pour un nombre de marques fixé. De plus, pour accélérer les opérations de recherche et de vérification, les auteurs de GARSP et de SHIFT ont utilisé des structures bitmaps. Ces structures sont des segments de bits dont chaque élément prend une valeur 1 ou 0 indiquant qu'une position est active ou non. Nous rappelons ci-dessous le principe de l'algorithme GARSP et nous commençons d'abord par rappeler les définitions des structures bitmaps et des vecteurs utilisés.

Supposons que nous désirons déterminer une règle optimale de  $M$  marques et que les bornes inférieure et supérieure notées respectivement **min-length** et **max-length** sont connues. À chaque itération, nous déterminons la position d'une nouvelle marque  $m$  en nous servant des structures de données. Cinq vecteurs sont utilisés dans GARSP. Ils sont définis ainsi :

Soit  $L$  la valeur de la borne supérieure **max-length**.

**LENGTH** est un vecteur de dimension  $M$  contenant les positions des marques, tel que  $\text{LENGTH}(1)=0$ .

$\text{LIST}(m)$  est une structure bitmap de dimension  $L$  qui contient les nouvelles différences mesurées après que la marque  $m$  est fixée, c'est-à-dire  $\text{LIST}(m)[\ell] = 1$  si  $\ell$  est une nouvelle distance mesurée et 0 sinon.

$\text{DIST}(m)$  est une structure bitmap de dimension  $L$  qui contient toutes les différences

mesurées par la règle courante de  $m$  marques, c'est-à-dire  $\text{DIST}(m)[\ell] = 1$  si la distance  $\ell$  est mesurée et 0 sinon.

$\text{COMP}(m)$  est une structure bitmap de dimension  $L$  qui contient toutes les différences interdites pour les prochaines marques à fixer. Nous utilisons  $\text{LENGTH}[m]$  comme origine, c'est-à-dire que  $\text{COMP}(m)[\ell] = 1$  si la position  $\ell + \text{LENGTH}[m]$  ne peut pas être occupée par la marque  $m + 1$ , où  $\ell$  est la différence entre les positions des marques  $m + 1$  et  $m$ .

$\text{FIRST}(2^{16})$  est un vecteur (tableau) de dimension  $2^{16}$  qui contient la première position du bit 0 pour les valeurs variant de 0 à  $2^{16} - 1$ , c'est-à-dire  $\text{FIRST}[p] = \ell$  où  $\ell$  est la position du premier bit égal à 0 dans la représentation binaire de  $p$ . Par conséquent, si  $p$  est l'entier dont la représentation binaire est  $\text{COMP}(m)$ ,  $\ell$  sera donc la première distance disponible pour la marque  $m + 1$ .

Nous présentons ci-dessous un exemple illustrant l'utilisation des différents vecteurs. Soient les quatre premières positions  $\{0,1,8,11\}$  suivantes d'une règle de  $M$  marques. Les distances mesurées sont  $\{1,3,7,8,10,11\}$ . Les nouvelles distances mesurées après avoir fixé la 4<sup>e</sup> marque sont  $\{3,10,11\}$  et les distances interdites pour les prochaines marques sont  $\{1,2,3,4,7,8,10,11\}$ . Voici la représentation de ces dernières sous forme de structure bitmaps.

$\text{DIST}(4)=10100011011000000000\dots$

$\text{LIST}(4)=00100000011000000000\dots$

$\text{COMP}(4)=11110011011000000000\dots$

Observons qu'en général, la longueur  $L$  est assez grande pour les règles avec un grand nombre de marques et il n'est pas toujours possible d'avoir des segments contigus de  $L$  bits. Par conséquent, les structures  $\text{COMP}(m)$ ,  $\text{LIST}(m)$  et  $\text{DIST}(m)$

seront représentées par  $\lceil L/32 \rceil$  segments de 32 bits, où  $\lceil L/32 \rceil$  est l'entier supérieur de la division de  $L$  par 32. De même, pour limiter la taille du vecteur FIRST, nous utilisons un tableau de taille  $2^{16}$ . Par conséquence, pour pouvoir utiliser  $\text{COMP}(m)$  comme index de FIRST, nous devons découper ce dernier en plusieurs segments de 16 bits. Par exemple si le premier bit 0 se trouve entre le 33<sup>e</sup> et le 48<sup>e</sup> bit, la distance  $\ell$  s'obtient de la manière suivante :

$$\ell = \text{FIRST}(A) \div 32, \text{ où } A = \text{COMP}(m)[1] \& \text{ffff0000},$$

c'est-à-dire en effectuant un ET LOGIQUE du vecteur COMP avec la représentation en base 16 d'un nombre dont les 16 premiers bits sont à 1 et 0 pour le reste. Ce nombre agit comme un masque pour isoler les 16 premiers bits de COMP. Rappelons qu'en représentation hexadécimale les bits sont regroupés par paquets de 4 et la lettre f correspond à la valeur hexadécimale dont tous les 4 bits sont à 1 (f = 1111).

Dans le cas où le premier bit 0 se trouve entre le 49<sup>e</sup> et le 64<sup>e</sup> bits du vecteur COMP, la formule suivante est utilisée :

$$\ell = \text{FIRST}(A) + 48, \text{ où } A = \text{COMP}(m)[1] \ll 16.$$

L'opérateur de décalage permet d'une part d'isoler les 16 derniers bits et d'autre part de ramener la valeur à une portée se trouvant à l'intérieur de la taille du vecteur FIRST.

Les structures bitmaps et les vecteurs utilisés facilitent les mises à jour à chaque itération et améliorent grandement l'efficacité de l'algorithme. Dans la partie qui suit, pour faciliter la compréhension de la mise à jour des structures bitmaps, nous supposons que les vecteurs COMP, LIST et DIST sont des vecteurs contigus de longueur  $L$ . De même que FIRST est un tableau de dimension  $2^L$  de sorte que nous puissions utiliser directement COMP comme indexage sans avoir à le découper en différents segments.

Supposons que nous venons de fixer la position de la marque  $m$ . La mise à jour de  $\text{LIST}(m)$  est obtenue par un décalage à droite de  $\text{LIST}(m-1)$  d'une distance  $\ell$ , où  $\ell$  est la distance entre les positions des marques  $m$  et  $m-1$  ( $\ell = \text{LENGTH}(m) - \text{LENGTH}(m-1)$ ). Par la suite nous forçons le  $\ell^{\text{ème}}$  bit de  $\text{LIST}(m)$  à 1, cela correspond à interdire la distance  $\ell$  pour les prochaines marques. La proposition suivante permet donc la mise à jour des vecteurs  $\text{LIST}$ .

**Proposition 1** *La mise à jour de  $\text{LIST}$  se fait en effectuant un décalage à droite de  $\ell = \text{LENGTH}(m) - \text{LENGTH}(m-1)$  des valeurs de l'itération précédente :*

$$\text{LIST}(m) \leftarrow \text{LIST}(m-1) \gg \ell$$

et en posant  $\text{LIST}(m)[\ell] = 1$ .

En considérant qu'il s'agit de codage en base 2, la formule de mise à jour peut se réécrire :

$$\text{LIST}(m) \leftarrow \text{LIST}(m-1) \gg (\text{LENGTH}(m) - \text{LENGTH}(m-1)) + 2^{\ell-1}.$$

*Preuve.* Supposons que nous venons de fixer la position de la marque  $m$ . Les nouvelles différences mesurées sont :

$$\text{LENGTH}(m) - \text{LENGTH}(k), \quad k = 0, 1, \dots, m-1.$$

En posant  $\ell = \text{LENGTH}(m) - \text{LENGTH}(m-1)$ , ces nouvelles distances se réécrivent :

$$\text{LENGTH}(m-1) + \ell - \text{LENGTH}(k), \quad k = 0, 1, \dots, m-1.$$

Cependant  $\text{LENGTH}(m-1) - \text{LENGTH}(k)$ , pour  $k = 0, 1, \dots, m-2$  sont les distances mesurées par  $\text{LIST}(m-1)$ . Par conséquence, les nouvelles valeurs de  $\text{LIST}(m)$  sont obtenues avec un décalage à droite de  $\ell$  positions des valeurs de  $\text{LIST}(m-1)$ , soit :

$$\text{LIST}(m) \leftarrow \text{LIST}(m-1) \gg \ell,$$



et en posant  $\text{LIST}(m)(\ell)=1$ .

Le vecteur  $\text{DIST}(m)$  qui rassemble toutes les distances mesurées par la règle après la fixation de la position  $m$ , correspond à des valeurs de  $\text{DIST}(m-1)$  et de  $\text{LIST}(m)$ , soit :

$$\text{DIST}(m) \leftarrow \text{DIST}(m-1) \vee \text{LIST}(m).$$

La mise à jour du vecteur  $\text{COMP}$  est décrite dans la proposition suivante :

**Proposition 2**

$$\text{COMP}(m) = \left[ \text{COMP}(m-1) \ll \left( \text{LENGTH}(m) - \text{LENGTH}(m-1) \right) \right] \vee \text{DIST}(m).$$

*Preuve.* Supposons que nous venons de fixer la position de la marque  $m$ . Les nouvelles distances mesurées sont :

$$\text{LENGTH}(m) - \text{LENGTH}(k), \quad k = 0, 1, \dots, m-1,$$

et sont mémorisées dans  $\text{LIST}(m)$ .

Déterminons les distances qui sont interdites pour les marques non encore fixées. Soit  $d$  une des nouvelles distances mesurées par la règle courante de  $m+1$  marques. Cela implique donc qu'il existe  $\tilde{m}$  tel que  $d = \text{LENGTH}(m) - \text{LENGTH}(\tilde{m})$ . Soit  $m'$  une marque différente de  $m$ . Nous démontrons qu'il n'est pas nécessaire de considérer les positions  $\text{LENGTH}(m') + d$  comme position interdite pour une marque, puisqu'il existe une nouvelle distance  $d''$  telle que  $\text{LENGTH}(m) + d'' = \text{LENGTH}(m') + d$ .

Si  $\text{LENGTH}(m') + d \leq \text{LENGTH}(m)$ , il n'est pas nécessaire de la considérer, puisque  $\text{LENGTH}(m)$  est la première position valide après la position  $\text{LENGTH}(m-1)$ .

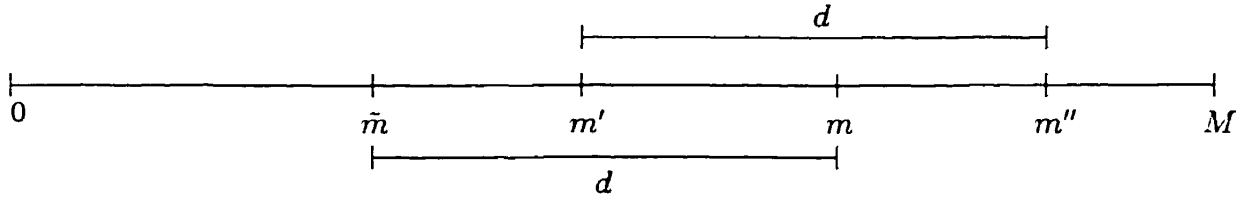


Figure 1.9 – Positions de marques

Si  $\text{LENGTH}(m') + d > \text{LENGTH}(m)$  considérons alors la position de la marque  $m''$  ainsi générée, c'est-à-dire  $m''$  tel que  $\text{LENGTH}(m'') = \text{LENGTH}(m') + d$ . Nous obtenons alors :

$$\begin{aligned} \text{LENGTH}(m'') - \text{LENGTH}(m') = \\ (\text{LENGTH}(m'') - \text{LENGTH}(m)) + (\text{LENGTH}(m) - \text{LENGTH}(m')). \end{aligned}$$

Cependant  $\text{LENGTH}(m'') - \text{LENGTH}(m) = \text{LENGTH}(m') - \text{LENGTH}(\tilde{m})$ , c'est-à-dire une distance qui a été déjà mesurée avant même qu'on ait déterminé la position de la marque  $m$ . La valeur de  $m''$  pourrait donc être trouvée en considérant toutes les distances (les anciennes et les nouvelles distances) appliquées à la marque  $m$ .

■

### 1.6.2 GARSP

Nous décrivons ci-dessous les principales étapes de GARSP déduites à partir de la description de l'algorithme SHIFT [9] et des commentaires dans le fichier source de GARSP (Rankin et Garry [32]). Il s'agit d'une méthode d'énumération implicite dans laquelle les marques sont fixées dans un ordre (stratégie de recherche) prédéfini. Dans GARSP, les marques sont fixées de gauche à droite (recherche en profondeur d'abord). D'autres stratégies seront examinées dans les sections qui suivent.

Dans GARSP, la borne supérieure **max-length** est fixée à la meilleure surestimation de la longueur de la règle optimale. Cette valeur est disponible dans la

littérature (tableaux 1.1 et 2.3). Au cours de la recherche sa valeur sera mise à jour dès qu'une nouvelle règle valide plus courte de  $M$  marques sera trouvée. Deux tests de réduction ont été considérés dans GARSP. Ces tests sont le test de la valeur milieu et le test de la valeur limite que nous définissons ci-dessous.

**Définition** : le test de la valeur milieu

Le test de la valeur milieu permet de réduire le domaine de la recherche en évitant de générer les règles dites symétriques. Observer qu'une partie de ces règles seront éliminées lorsque la marque milieu est limitée à une position inférieure à la moitié de la longueur de la règle courante.

**Définition** : le test de la valeur limite

Ce test permet de limiter la position que peut occuper une marque  $m$  étant donné la longueur **max-length** de la règle courante. Soit  $\mathbf{golomb}_{\text{DIST}(m)}(M - m + 1)$  la longueur optimale d'une règle de  $M - m + 1$  marques pour laquelle les valeurs de  $\text{DIST}(m)$  sont interdites, c'est-à-dire pour laquelle les distances de  $\text{DIST}(m)$  ne sont pas mesurées par la règle en question. Ces longueurs sont enregistrées dans une base de données  $\mathcal{D}$  qu'on consultera au besoin. Le test indique que :  $\text{LENGTH}(m)$  ne doit pas être supérieure à **max-length** -  $\mathbf{golomb}_{\text{DIST}(m)}(M - m + 1)$  si nous voulons obtenir une solution qui soit meilleure que la solution courante.

L'idée de l'algorithme GARSP est de choisir à chaque itération une position disponible pour une marque  $m$  à l'aide des vecteurs COMP et FIRST tout en respectant le test de la valeur milieu et de la valeur limite. Une itération est définie comme l'ensemble des opérations effectuées pour fixer une marque quelconque, c'est-à-dire trouver une position valide pour une marque et mettre à jour les différentes struc-

tures bitmaps. Si une position valide est trouvée, les vecteurs LIST, DIST et COMP seront mises à jour et nous fixerons la prochaine marque, sinon un retour arrière sera effectué et une nouvelle position sera déterminée pour la marque  $m - 1$ . Les solutions sont imprimées au fur et à mesure qu'on les trouve et la recherche s'arrête lorsque l'algorithme est en mesure de conclure que la règle courante est optimale.

### Procédure GARSP

La procédure décrite ci-dessous correspond à l'algorithme de dérivation des règles de Golomb optimales de Rankin et Garry telle que décrite dans [32].

- 1) Initialiser les structures bitmaps :
- 2)  $LIST(1) \leftarrow 1; DIST(1) \leftarrow 0; COMP(1) \leftarrow 0; LENGTH(1) = 0; m \leftarrow 2;$
- 3) Initialiser la borne supérieure **max-length**.
- 4) Déterminer la position de la marque  $m$  :
- 5)  $\ell \leftarrow FIRST(COMP(m - 1));$
- 6) Fixer la position de la marque  $m$  :
- 7)  $LENGTH(m) \leftarrow LENGTH(m - 1) + \ell;$
- 10) Test de la valeur limite :
- 11) **Si**  $LENGTH(m) < \text{max-length} - \text{golomb}_{DIST(m-1)}(M - m + 1)$  **alors**
- 12) Test de la valeur milieu :
- 13) **Si**  $m = \frac{M}{2}$  **et**  $LENGTH(m) < \frac{\text{max-length}}{2}$  **alors**
- 14) **Si**  $m < M$  **alors**
- 15) Mettre à jour les structures LIST, DIST, COMP.
- 16)  $LIST(m) \leftarrow (LIST(m - 1) \gg \ell) + 2^{\ell-1};$
- 17)  $DIST(m) \leftarrow DIST(m - 1) \vee LIST(m);$
- 18)  $COMP(m) \leftarrow (COMP(m - 1) \ll \ell) \vee DIST(m);$
- 19)  $m \leftarrow m + 1;$

```

20)                Aller à la ligne 4.
21)                Sinon
22)                Sauvegarder la règle trouvée.
23)                Initialiser max-length.
24)                 $m \leftarrow m - 1$ ;
25)                Si  $m = 1$  Alors
26)                    Arrêter : on a trouvé une règle optimale.
27)                Finsi;
28)                Finsi;
29)                Sinon
30)                     $m \leftarrow m - 1$ ;
31)                Finsi;
32)                Si  $m = 1$  Alors
33)                    Arrêter : on a trouvé une règle optimale.
34)                Finsi;
35)                Sinon
36)                     $m \leftarrow m - 1$ ;
37)                Si  $m = 1$  Alors
38)                    Arrêter : on a trouvé une règle optimale.
39)                Finsi;
40)                Finsi;

```

**Théorème 1** *Chaque itération de GARSP est de complexité  $O(L/16)$ .*

*Preuve.* Supposons que nous cherchons une position  $\ell$  pour la marque  $m$ . Cette position est obtenue en se servant des structures COMP et FIRST (un tableau de taille  $2^{16}$ ), c'est-à-dire  $\ell = \text{FIRST}(\text{COMP}(m))$ . Dans le pire des cas, le premier bit 0 se retrouve dans les 16 derniers bits du vecteur COMP, nous devons donc passer à travers les  $\lceil L/16 \rceil$  segments de  $\text{COMP}(m)$ . Cette opération nécessite  $O(L/16)$ .

D'autre part, nous devons aussi mettre à jour les structures  $\text{COMP}(m)$ ,  $\text{LIST}(m - 1)$ ,  $\text{DIST}(m - 1)$  et initialiser  $\text{DIST}(m)$ ,  $\text{LIST}(m)$  et  $\text{COMP}(m + 1)$ . Chacune de ces opérations nécessite  $O(1)$  opérations. La complexité d'une itération est donc  $O(L/16)$ . ■

## 1.7 Algorithme de recherche heuristique : plans projectifs et semi-affines

Nous décrivons ici le principe des meilleurs algorithmes heuristiques pour la recherche d'une règle de Golomb optimale.

### 1.7.1 Définitions

L'algorithme de dérivation des règles de Golomb par construction de plans repose sur la notion des ensembles de différences cycliques. Cette procédure, décrite par Coolsaet [36], consiste à construire des **plans semi-affines** ou des **plans projectifs** que nous définissons ci-dessous.

**Définition** : plan semi-affine

Un plan semi-affine est un ensemble de points et de droites dans un espace semi-linéaire qui doit satisfaire les propriétés suivantes :

1. Deux points sont reliés par au plus une droite.
2. Deux droites s'intersectent en au plus un point.
3. Étant donné une droite et un point qui n'est pas sur cette droite, toutes les autres droites passant par ce point doivent intersecter la première droite, à l'exception d'une droite.
4. Étant donné un point et une droite ne traversant pas ce point, tous les points de la droite doivent être connectés au premier point sauf un.

L'ordre d'un plan, noté  $q$ , est le nombre de points sur chaque droite du plan. Un plan semi-affine d'ordre  $q$  aura  $q^2 - 1$  points et  $q^2 - 1$  droites. Chaque point est traversé par exactement  $q$  droites. Nous verrons par la suite que l'ordre du plan

correspond à la taille  $s$  (nombre de marques de la règle de Golomb déduite de l'ensemble de différences cycliques) de l'ensemble de différences cycliques associé à ce plan.

**Définition : plan projectif**

Un plan projectif est un ensemble de points et de droites dans l'espace semi-linéaire qui doit satisfaire les deux propriétés suivantes :

1. Deux points sont reliés par exactement une et une seule droite.
2. Deux droites s'intersectent toujours en un seul point.

### 1.7.2 Construction des ensembles de différences cycliques

Nous décrivons ci-dessous une approche de dérivation des ensembles de différences cycliques par construction de plans. L'idée est de construire une droite, c'est-à-dire déterminer l'ensemble des points qui sont sur une droite du plan semi-affine ou projectif. Pour cela, nous faisons appel aux critères suivants :

**Critère A : plan semi-affine**

Soient  $(x, y)$  les coordonnées associées à un point,  $[u, v]$  les coordonnées associées à une droite du plan semi-affine, avec  $x, y, u$  et  $v$  compris entre  $[0, q - 1]$  où  $q$  est l'ordre du plan. La coordonnée  $(0,0)$  est interdite pour les points et les droites. Nous affirmons que le point  $(x, y)$  est sur la droite  $[u, v]$  si la relation suivante est vérifiée :

$$x * u + y * v = 1 \quad (q),$$

c'est-à-dire le produit terme à terme  $(x * u + y * v)$  modulo  $q$  est égal à 1. Il est clair que si nous associons le couple  $(0,0)$  à un point, ce point n'appartiendra à aucune droite et s'il est associé à une droite, aucun point ne sera sur cette droite. C'est



pourquoi cette coordonnée est interdite pour les points et les droites.

### Critère B : plan projectif

Soient  $(x, y, z)$  les coordonnées associées à un point,  $[u, v, w]$  les coordonnées associées à une droite du plan projectif avec  $x, y, z, u, v$  et  $w$  compris entre  $[0, q - 1]$  où  $q$  est l'ordre du plan. De même, le triplet  $(0,0,0)$  est interdit comme coordonnées des droites et des points. Deux coordonnées représentent le même point ou la même droite si l'une est produit scalaire de l'autre, par exemple  $(1,1,1)$  et  $(2,2,2)$  correspondent au même point ou à la même droite. Par définition, le point  $(x, y, z)$  est sur la droite  $[u, v, w]$  si la relation suivante est vérifiée :

$$x * u + y * v + z * w = 0 \quad (q),$$

c'est-à-dire que le produit  $(x * u + y * v + z * w)$  modulo  $q$  est nul. Observons que si les coordonnées  $(0,0,0)$  sont utilisées, le point  $(0,0,0)$  appartiendra à toutes les droites du plan et tous les points sont sur la droite  $(0,0,0)$ .

Une fois la droite construite, nous devons numéroter les points de telle sorte qu'on obtienne un ensemble de différences cycliques. Pour cela nous utilisons les séquences calculées à partir d'équations de récurrences obtenues par exemple avec la suite de Fibonacci. Les équations de récurrence sont des équations qui permettent de générer des suites de valeur telle que le  $n^{\text{ème}}$  élément est calculé à partir des  $m$  éléments précédents. Par exemple  $a_1 = 0, a_2 = 1$  et  $a_i = A * a_{i-1} + B * a_{i-2}$ , pour tout  $i \geq 2$ , est une équation de récurrence. Les paramètres  $A$  et  $B$  sont des scalaires quelconques. On applique un modulo  $q$  aux éléments de la séquence pour obtenir une suite périodique. La raison de cette opération est de permettre par la suite d'associer la nouvelle séquence aux coordonnées des points du plan. Le nombre de valeurs avant la première répétition sera la période de la séquence et sera utilisé comme module  $n$  de l'ensemble de différences cycliques. La taille  $s$  de l'ensemble de



À partir de cet ensemble, nous pouvons calculer la table de différences ci-dessous :

	0	3	4	17	19	(24)
0	0	3	4	17	19	
3	21	0	1	14	16	
4	20	23	0	13	15	
17	7	10	11	0	2	
19	5	8	9	22	0	

et nous déduisons les 5 règles de Golomb suivantes :

$\{0, 3, 4, 17, 19\}$ ,  $\{0, 1, 14, 16, 21\}$ ,  $\{0, 13, 15, 20, 23\}$ ,  
 $\{0, 2, 7, 10, 11\}$ ,  $\{0, 5, 8, 9, 22\}$ .

La méthode proposée est assez simple, cependant le grand défaut est de trouver la bonne équation de récurrence avec une période suffisamment grande pour la taille  $s$  désirée. De plus, l'auteur précise que la méthode ne fonctionne que pour un nombre  $s$  qui est premier. Nous décrivons ci-dessous un pseudocode simple de calcul des ensembles de différences cycliques du programme SEMIAFF.CC récupéré du site de Coolsaet [37]. Cette procédure utilise une famille d'équations de récurrence du type  $a_1 = 0, a_2 = 1$  et  $a_i = A * a_{i-1} + B * a_{i-2}$  où  $A$  et  $B$  sont des entiers quelconques. Les coordonnées de la droite choisie sont  $(0,1)$  et les positions de l'ensemble de différences cycliques sont enregistrées dans EDC, un vecteur de dimension  $s$ .

### Fonction RECURRENCE

- 1) Initialisation :
- 2)  $post \leftarrow 0; cnt \leftarrow 0; prec \leftarrow 1; last \leftarrow A;$
- 3) Calculer la séquence et déterminer l'ensemble de différences cycliques :
- 4) **Tant que**  $prec \neq 0$  ou  $last \neq 1$  **faire**

- 5)     **Si**  $prec = 1$  **alors**
- 6)          $EDC(cnt) \leftarrow post;$
- 7)          $cnt \leftarrow cnt + 1;$
- 8)     **Finsi** ;
- 9)      $tmp \leftarrow (A * last + B * prec) \bmod q;$
- 10)      $prec \leftarrow last;$
- 11)      $last \leftarrow tmp;$
- 12)    **Finpour** ;
- 13) retourner la période  $(post + 1)$ .

### **Procédure SEMIAFF**

- 1) Initialisation :
- 2)      $Trouver \leftarrow faux; A \leftarrow 1; B \leftarrow 1;$
- 3) Lire l'ordre  $q$  du plan semi-affine.
- 4) Trouver la bonne séquence de récurrence et l'ensemble de différences cycliques.
- 5)     **Tant que**  $A < q$  et  $Trouver = faux$  **faire**
- 6)         **Tant que**  $B \leq A$  et  $Trouver = faux$  **faire**
- 7)             Appeler la fonction **RECURRENCE** :
- 8)              $Periode \leftarrow RECURRENCE(q, A, B, EDC);$
- 9)             **Si**  $Periode = q^2 - 1$  **alors**
- 10)                 Imprimer l'ensemble de différences cycliques.
- 11)             **sinon**
- 12)                  $A \leftarrow A + 1; B \leftarrow B + 1;$
- 13)             **Finsi** ;
- 14)     **Finpour** ;
- 15) **Finpour** ;

L'idée du programme consiste principalement à déterminer les constantes  $A$  et  $B$  de telle sorte que la séquence modulée ait la bonne période pour l'ordre  $q$ . Le choix de la droite associée aux coordonnées  $(0,1)$  facilite le calcul des points sur cette dernière. En ce faisant, trouver la position des points qui vérifie le critère  $A$  revient simplement à déterminer la position des 1 dans la séquence modulée.

## CHAPITRE 2

### VERSIONS AMÉLIORÉES DE L'ALGORITHME GARSP

#### 2.1 Introduction

Nous proposons dans cette section une version améliorée de l'algorithme GARSP sous différents aspects que nous détaillons ci-dessous. Nous proposons par la suite une deuxième version exploitant un nouvel ordre de recherche.

#### 2.2 G-DFS1

Le défaut avec le test de la valeur milieu est que si nous ne connaissons pas la valeur optimale d'une règle de  $M$  marques et que nous commençons avec une valeur supérieure à **golomb**( $M$ ), il est fort possible que nous considérions plusieurs règles symétriques (voir sous-section 2.5.1) dans la recherche. Une façon de contourner ce défaut consiste à imposer que la première différence soit supérieure à la dernière différence. Ce test fonctionne parce que **GARSP** commence par fixer la première marque en commençant par une des extrémités de la règle. Nous pouvons alors utiliser le test suivant :

*Test de réduction 1 : Élimination des règles symétriques*

**Si**  $m = M - 1$  et  $\left(\text{LENGTH}(M) - \text{LENGTH}(M - 1)\right) < \text{LENGTH}(1)$  **alors**

Effectuer un retour-arrière.

**Finsi ;**

En utilisant le test de réduction 1 et en observant que d'autres réductions seraient possibles en considérant que  $\text{LENGTH}(2) + \left(\text{LENGTH}(M) - \text{LENGTH}(M - 1)\right) + \text{golomb}(M - 3) < \text{max-length}$ , nous obtenons le test suivant :

Si  $m = 1$  ET  $2 * \text{LENGTH}(2) + 1 > \text{max-length} - \text{golomb}(M - 3)$  ALORS

Effectuer un retour-arrière.

Finsi ;

Le dernier test que nous considérons porte sur le calcul d'une borne inférieure qui indiquera la meilleure règle qu'il serait possible d'obtenir avec la règle courante de  $m$  marques. Si cette borne excède la longueur de la règle courante de  $M$  marques, il n'est plus nécessaire de poursuivre les recherches. Nous pouvons alors effectuer un retour arrière à un niveau supérieur et considérer une autre direction de recherche. Cela est possible en considérant le test suivant :

*Test de réduction 3 : borne inférieure*

Calculer la borne inférieure **min-length** ;

Si **min-length** > **max-length** ALORS

Effectuer un retour arrière.

Finsi ;

Le raisonnement ci-dessus est aussi valide lorsque la première marque est obtenue par un retour arrière. En d'autres termes, il existe une position pour  $m$  supérieure à la valeur courante de  $\text{LENGTH}(m)$  qui pourrait aboutir à une règle plus courte. Cependant aucune répétition n'a été utilisée pour la position  $\text{LENGTH}(m)$ .

## 2.3 Stratégie de recherche - DFS

Pour un même algorithme, différentes stratégies de recherche peuvent être utilisées pour déterminer les règles. Nous proposons d'examiner deux de ces stratégies

dans cette section. Notons les par *ordre 1* et *ordre 2*. Soient  $a_1, a_2, \dots, a_{M-1}$  les différences mesurées entre les marques  $\{1,2\}, \{2,3\}, \dots, \{M-1,M\}$ , c'est-à-dire entre deux marques consécutives d'une règle de  $M$  marques.

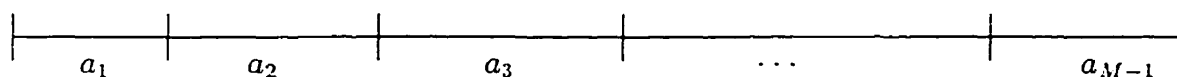


Figure 2.1 – Numérotation des différences des marques consécutives

L'*ordre 1* consiste à fixer les marques  $a_1, a_2, \dots, a_{M-1}$  de gauche à droite. Par défaut, c'est la stratégie privilégiée dans la plupart des algorithmes de la littérature, incluant GARSP.

L'*ordre 2* explore la recherche d'une règle en examinant les deux extrémités. Les marques sont fixées en alternant de gauche à droite. Nous commençons donc par fixer  $a_1$  ensuite  $a_{M-1}$ ,  $a_2$  suivi de  $a_{M-2}$  et ainsi de suite. Cette stratégie permet de développer une nouvelle procédure de recherche que nous notons G-DFS2.

Par contre, de légères modifications doivent être apportées dans G-DFS2. Étant donné que nous cherchons une règle optimale, à priori la longueur de la règle est inconnue, il n'est donc pas possible de déterminer la position exacte des marques avant d'avoir trouvé la règle au complet. Pour remédier à ce problème nous travaillons avec les différences entre les marques consécutives, c'est-à-dire au lieu de déterminer la position des marques, nous allons déterminer les  $M - 1$  différences entre les marques consécutives de la règle.

De même, les mises à jour du vecteur COMP seront aussi modifiées. Les distances interdites, c'est-à-dire le vecteur  $\text{COMP}(m + 1)$  sera obtenu, à chaque itération, par la proposition suivante :



**Proposition 3** Soient  $a_1, a_2, \dots, a_{m-1}$  les différences mesurées entre les paires de marques consécutives telles que  $a_i = \text{LENGTH}(i+1) - \text{LENGTH}(i)$ , pour  $i = 1, 2, \dots, m$ . Les distances interdites ( $\text{COMP}(m+1)$ ) sont :

$$\forall i = 1, 2, \dots, m \quad \text{COMP}(m+1) = \text{COMP}(m+1) \vee \text{DIST}(m) \vee \left( \text{DIST}(m) \ll \sum_{j=i}^m a_j \right).$$

*Preuve.* Considérons  $\text{DIST}(m)$  l'ensemble des distances mesurées et  $a_1, a_2, \dots, a_{m-1}$  les différences consécutives entre les  $m$  premières marques. Déterminons les nouvelles distances  $d$  qui sont interdites suite à l'affectation de la marque  $m$  à la position  $\text{LENGTH}(m)$ . Soit  $m'$  la prochaine marque que nous désirons fixer. Observer que toute position de  $m'$  telle que  $\text{LENGTH}(m') - \text{LENGTH}(i)$  est une distance comprise dans  $\text{DIST}(m)$  est une position interdite pour la nouvelle marque  $m'$ .

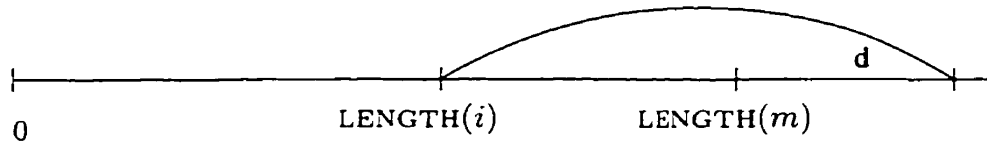


Figure 2.2 – Distances interdites

Par conséquent, les nouvelles distances interdites  $d$  sont telles que :

$$\forall i = 0, 1, 2, \dots, m-1 \quad d + \text{LENGTH}(m) - \text{LENGTH}(i) \in \text{DIST}(m),$$

où encore :

$$\forall i = 0, 1, 2, \dots, m-1 \quad d \in \text{DIST}(m) - (\text{LENGTH}(m) - \text{LENGTH}(i)).$$

Cependant

$$\forall i = 0, 1, 2, \dots, m-1 \quad \text{LENGTH}(m) - \text{LENGTH}(i) = \sum_{j=i}^m a_j.$$

Nous pouvons alors déterminer toutes les nouvelles distances  $d$  interdites en effectuant :

$$\forall i = 1, 2, \dots, m \quad \text{DIST}(m) \ll \sum_{j=i}^m a_j.$$

Finalement, l'ajout de ces valeurs au vecteur  $\text{COMP}(m+1)$  constitue toutes les distances interdites pour les prochaines marques. Étant donné que nous calculons la règle en déterminant les deux segments qui la composent, deux initialisations du vecteur  $\text{LIST}$  sont nécessaires, le premier pour les nouvelles distances générées par le segment de gauche et l'autre pour les nouvelles distances générées par le segment de droite. Cependant leurs mises à jour, de même que celle du vecteur  $\text{DIST}$  demeurent inchangées.

**Exemple :** Calcul des distances interdites

Supposons que nous ayons fixé les quatre premières marques d'une règle de  $M$  marques aux positions suivantes.

$$0 \quad 1 \quad 10 \quad 13,$$

les différences entre les marques consécutives sont :

$$a_1 = 1 \quad a_2 = 9 \quad a_3 = 3,$$

et les distances mesurées sont :

$$\text{DIST}(3) = \{1, 3, 9, 10, 12, 13\}.$$

Déterminons alors les distances qui sont interdites pour les prochaines marques à fixer.

$$\text{DIST}(3) \ll a_3 \rightarrow \{6, 7, 9, 10\}.$$

$$\text{DIST}(3) \ll a_3 + a_2 \rightarrow \{1\}.$$

$$\text{DIST}(3) \ll a_3 + a_2 + a_1 \rightarrow \{\}.$$

Par conséquence, les distances interdites sont :

$$\text{COMP}(4) = \{1, 3, 6, 7, 9, 10, 12, 13\}.$$

## 2.4 G-DFS2

Nous détaillons ci-dessous l'algorithme de dérivation des règles de Golomb optimales utilisant l'ordre 2 comme stratégie de recherche. Les indices de valeurs paires du vecteur `LIST` correspondent aux distances générées par le segment de gauche et les indices de valeurs impaires à celles générées par le segment de droite. De plus les distances entre les marques consécutives sont enregistrées dans le vecteur `COUNT`, c'est-à-dire que  $\text{COUNT}(m) = \text{LENGTH}(m) - \text{LENGTH}(m - 1)$ .

- 1) Initialiser les structures bitmap.
- 2)  $\text{LIST}(0) \leftarrow 1; \text{LIST}(1) \leftarrow 1; \text{DIST}(0) \leftarrow 0; \text{COMP}(0) \leftarrow 0; \text{COMP}(1) \leftarrow 0$
- 3)  $m \leftarrow 2;$
- 4) Initialiser les bornes supérieure **max-length** et inférieure **min-length**.
- 5) Déterminer la valeur  $\ell$  pour  $a_{m-1}$ .
- 6) **Faire**
- 7)  $\ell \leftarrow \text{FIRST}(\text{COMP}(m - 1));$
- 8)  $\text{COMP}(m - 1) \leftarrow \text{COMP}(m - 1) + 2^{\ell-1};$
- 9) **Tant** que  $m = M$  et  $\text{COUNT}(1) > \ell;$
- 10) Fixer la valeur de  $a_m$ .
- 11) **Si**  $(m \bmod 2) = 1$  **alors**
- 12)  $\text{COUNT}(\frac{m}{2}) \leftarrow \ell;$
- 13) **Sinon**
- 14)  $\text{COUNT}(M - \frac{m-1}{2}) \leftarrow \ell;$
- 15) **Finsi**;

- 16)      $\text{LENGTH} = \text{LENGTH} + \ell;$
- 17) Calculer la nouvelle borne inférieure **min-length**.
- 18)      $\text{min-length} \leftarrow \text{LENGTH}(m) + \text{golomb}_{\text{DIST}}(M - m + 1);$
- 19) Test de la borne inférieure.
- 20)     **Si min-length < max-length alors**
- 21)         **Si  $m = M$  alors**
- 22)             Sauvegarder la règle trouvée.
- 23)             Initialiser **max-length**.
- 24)              $m \leftarrow m - 1;$
- 25)     **Sinon**
- 26)     Mettre à jour les structures LIST, DIST, COMP.
- 27)         **Si  $(m \bmod 2) = 1$  alors**
- 28)              $\text{LIST}(m) \leftarrow \left( \text{LIST}(m - 2) \gg \ell \right) + 2^{\ell-1};$
- 29)              $\text{DIST}(m) \leftarrow \text{DIST}(m - 1) \vee \text{LIST}(m);$
- 30)              $\text{COMP}(m + 1) \leftarrow \text{DIST}(m);$
- 31)              $d \leftarrow 0;$
- 32)             **Pour  $i = \frac{m-1}{2}$  à 0 faire**
- 33)                  $d \leftarrow d + \text{COUNT}(i);$
- 34)                  $\text{COMP}(m + 1) \leftarrow \text{COMP}(m + 1) \vee \left( \text{DIST}(m + 1) \ll d \right);$
- 35)             **FinPour ;**
- 36)     **Sinon**
- 37)              $\text{LIST}(m) \leftarrow \left( \text{LIST}(m - 2) \gg \ell \right) + 2^{\ell-1};$
- 38)              $\text{DIST}(m) \leftarrow \text{DIST}(m - 1) \vee \text{LIST}(m);$
- 39)              $\text{COMP}(m + 1) \leftarrow \text{DIST}(m);$
- 40)              $d \leftarrow 0;$
- 41)             **Pour  $i = M - \frac{m}{2}$  à M-1 faire**
- 42)                  $d \leftarrow d + \text{COUNT}(i);$
- 43)                  $\text{COMP}(m + 1) \leftarrow \text{COMP}(m + 1) \vee \left( \text{DIST}(m + 1) \ll d \right);$

```

34)          FinPour ;
45)          FinSi ;
46)           $m \leftarrow m + 1 ;$ 
47)          Aller à la ligne 4.
48)          Finsi ;
49)  Sinon
50)           $m \leftarrow m - 1 ;$ 
51)          Si  $m = 1$  Alors
52)              Arrêter : on a trouvé une règle optimale.
53)          Finsi ;
54)  Finsi ;

```

Nous présentons ci-dessous un exemple de dérivation d'une règle de Golomb de 5 marques. Nous effectuons les premières itérations jusqu'à l'obtention d'une première règle afin de mettre en évidence les différentes opérations de mise à jour des structures bitmaps.

1) Initialiser les structures bitmap.

$LIST(0) = \{1\};$

$LIST(1) = \{1\};$

$DIST(0) = \{ \};$

$COMP(0) = \{ \};$

$COMP(1) = \{ \};$

2) 1<sup>ère</sup> itération.

$m \leftarrow 2;$

$COUNT(\frac{m}{2}) = 1 ;$

$LIST(m) = \{1\};$

$$\text{DIST}(m) = \{1\};$$

$$\text{COMP}(m) = \{1\};$$

3) 2<sup>e</sup> itération.

$$m \leftarrow 3;$$

$$\text{COUNT}(M - \frac{m-1}{2}) = 2;$$

$$\text{LIST}(m) = \{2\};$$

$$\text{DIST}(m) = \{1, 2\};$$

$$\text{COMP}(m) = \{1, 2\};$$

4) 3<sup>e</sup> itération.

$$m \leftarrow 4;$$

$$\text{COUNT}(\frac{m}{2}) = 3;$$

$$\text{LIST}(m) = \{3, 4\};$$

$$\text{DIST}(m) = \{1, 2, 3, 4\};$$

$$\text{COMP}(m) = \{1, 2, 3, 4\};$$

5) 4<sup>e</sup> itération.

$$m \leftarrow 5;$$

$$\text{COUNT}(M - \frac{m-1}{2}) = 2;$$

$$\text{LIST}(m) = \{5, 7\};$$

$$\text{DIST}(m) = \{1, 2, 3, 4, 5, 7\};$$

$$\text{COMP}(m) = \{1, 2, 3, 4, 5, 7\};$$

6) Imprimer la solution

trouver : longueur = 11

0	1	4	9	11
1	3	5	2	

## 2.5 Tests de réduction

Les tests de réduction sont utilisés pour restreindre le domaine de recherche. Dans le cadre des règles de Golomb, nous en considérons trois. Un premier test simple pour éliminer les règles symétriques, un deuxième pour limiter la position qu'une marque peut occuper et un dernier pour estimer la longueur minimale éventuelle de la règle.

### 2.5.1 Élimination des règles symétriques

Ce test a pour but d'éliminer la génération des règles symétriques. Une première solution consiste à fixer la deuxième marque à la position 2 (Dewdney[7]). Cependant cette approche ne permet pas de toutes les éliminer. Une autre solution consiste à limiter la position de la marque du milieu à la moitié de la longueur de la règle courante ( $\text{LENGTH}(m) < \mathbf{max-length}/2$  pour  $m \leq \lceil \frac{M}{2} \rceil$ ). Ce test est utilisé dans GARSP, c'est ce que nous appelons le test de la valeur milieu. Cependant, le même problème persiste, nous n'obtenons qu'une réduction partielle. Par exemple, une règle optimale de 9 marques mesure une longueur de 44 unités. L'ensemble des différences entre les marques consécutives sont :  $\{3,6,8,2,3,7,4,1\}$ . Observer que la valeur milieu est de 22 unités ce qui est supérieure à 19 et à 15 qui sont respectivement la somme des 4 premières ( $\{3,6,8,2\}$ ), et des 4 dernières ( $\{3,7,4,1\}$ ) différences des marques consécutives de la règle. Par conséquence, les deux règles seront générées lors de la recherche si le test de la valeur milieu est utilisé. Par contre, observer que toutes les règles symétriques seront éliminées en imposant  $\text{LENGTH}(2) < \text{LENGTH}(M) - \text{LENGTH}(M - 1)$  lorsqu'on utilise GARSP.

### 2.5.2 Test de la position limite

L'utilisation d'une borne supérieure permet de limiter la position maximale que peut occuper une marque. Soit **max-length** une borne supérieure pour une règle

de  $M$  marques et  $\text{golomb}(N)$  la longueur optimale d'une règle de  $N$  marques. La position maximale que peut occuper une marque  $m$  peut être obtenue par  $\text{LIMITE} = \text{max-length} - \text{golomb}(M - m + 1)$  (Dollas, Rakin et McCracken[9]). Il est facile de constater que si  $m$  est la marque courante à fixer, sans inclure cette dernière, il restera  $M - m + 1$  marques à déterminer. Par conséquent, si  $\text{LENGTH}(m) + \text{golomb}(M - m + 1) > \text{max-length}$ , nécessairement la règle que nous trouverons sera moins bonne que la règle courante puisque  $\text{golomb}(M - m + 1)$  est la plus petite longueur que nous puissions avoir pour une règle valide de  $M - m + 1$ . Ainsi cette limite doit être respectée si nous voulons améliorer la solution.

Observer que si certaines valeurs ou différences sont interdites, c'est-à-dire qu'elles sont mesurées par la règle partielle courante de  $m$  marques, la valeur limite pourrait être réduite davantage en considérant  $\text{LIMITE} = \text{max-length} - \text{golomb}_{\text{DIST}(m-1)}(M - m + 1)$  où  $\text{golomb}_{\text{DIST}(m-1)}(M - m + 1)$  est la longueur optimale d'une règle de  $M - m + 1$  marques telle que les distances de  $\text{DIST}(m - 1)$  sont interdites.

### 2.5.3 Test de la borne inférieure

#### 2.5.3.1 Définition de la borne inférieure

Le dernier test que nous considérons porte sur la borne inférieure. Il permet de vérifier si la recherche aboutira à une meilleure solution compte tenu de la position des marques déjà fixées. Plusieurs méthodes de calcul des bornes inférieures ont été proposées dans la littérature. Une première consiste à fixer cette valeur à  $\frac{M(M-1)}{2}$ , la longueur d'une règle parfaite de  $M$  marques. Puisqu'une règle de Golomb parfaite mesure toutes les différences entre 0 et sa longueur, cette longueur constitue une borne inférieure pour la règle de  $M$  marques. En 1986 [1], Atkinson *et al.* ont proposé une amélioration sur cette borne. En 1991 [23], Lorentzen et Nilsen proposent



une formulation linéaire du problème permettant d'obtenir de meilleures bornes inférieures. D'autres bornes inférieures sont fournies par Linbarger, Sudborough et Tollis [22], Chen et Kløve [6], Kløve [18] [19] [20] et Hansen, Jaumard et Meyer [16].

Dans ce mémoire nous en considérons une (Hansen, Jaumard et Meyer [16]) qui conduit à une borne équivalente à celle qu'obtient Kløve [19] dans le cas des triangles de différences (DTS) et qui est la meilleure borne connue aujourd'hui. La méthode considérée consiste à déterminer un ensemble de chemins dans un graphe complet du noeud 0 au noeud  $M - 1$  qui donnera une longueur maximale à la borne inférieure. Le choix de cette approche repose principalement sur la qualité de la borne fournie.

**Proposition 4** (Hansen, Jaumard et Meyer [16]) Soient  $m$  une marque nouvellement fixée et  $\ell_{ij} = \text{length}(i) - \text{length}(j)$ , pour  $j < i \in \{1, 2, \dots, m\}$  et  $\ell_{pq} = \{\text{plus petites valeurs disponibles}\}$  pour  $p \in \{m + 1, m + 2, \dots, M\}$  et  $q \in \{m - k + 1, m - k + 2, \dots, m\}$ , avec  $k \in \{1, \frac{M}{2}\}$  tel que la valeur de la borne inférieure  $\underline{S}$  est maximale, nous obtenons :

$$\frac{k(k+1)}{2} \underline{S} = \sum_{i=2}^M \sum_{j=i-k}^{i-1} \ell_{ij} + \sum_{i=2}^k w_{i,i-1}^k (\ell_{i,i-1} + \ell_{M-i+2, M-i+1}), \quad (2.1)$$

où  $w_{i,i-1}^k = \frac{(k-i+1)(k-i)}{2}$  est le poids de la différence.

La proposition ci-dessus permet de calculer une borne inférieure à chaque itération. En pratique, les valeurs  $\ell_{ij}$  coûtent cher en temps de calcul à déterminer lorsque certaines marques sont déjà fixées. Nous sommes obligés de trouver un compromis entre la qualité de la borne et le coût de calcul. Cependant si aucune marque n'est encore fixée, la borne inférieure peut être obtenue analytiquement.

**Proposition 5** (Kløve [19] et Chen et Kløve [6]) Soit  $M$  le nombre de marques et  $1 \leq k \leq \frac{M}{2}$ , qui donne la meilleure borne inférieure. La formule suivante permet de

*calculer une borne inférieure de même qualité de celle obtenue dans la proposition 4.*

$$\underline{S}^0 = \frac{k}{k+1}M^2 + \frac{-k^2 - k + 1}{k+1}M + \frac{1}{12}(k-1)(7k+10).$$

Soit **min-length**, une borne inférieure obtenue après avoir fixé la marque  $m$ . Nous pouvons réduire le domaine de recherche en imposant que **min-length** < **max-length**. Ce test assure que nous améliorons toujours la solution.

### 2.5.3.2 Calcul de la borne inférieure

Les meilleures bornes inférieures sont souvent les plus complexes à calculer. Pour les exploiter pleinement, nous devons développer des méthodes efficaces pour les calculer. Nous proposons dans cette section une méthode pour y parvenir. Nous définissons dans cette section  $\text{PARTIAL-BOUND}(m)$  la somme de toutes les différences considérées dans la borne inférieure. Par exemple, si  $m$  est la marque nouvellement fixée, alors  $\text{PARTIAL-BOUND}(m)$  contiendra la somme de toutes les  $\ell_{ij}$  pour  $i \in \{1, 2, \dots, m\}$  et pour  $j \in \{i-k, i-k+1, \dots, i\}$ .

### 2.5.3.3 Borne inférieure dans G-DFS1

Examinons désormais les étapes nécessaires pour calculer la borne inférieure dans l'ordre 1. Tel qu'il a été mentionné, la proposition 4 peut être utilisée pour calculer cette borne à chaque itération. Pour cela, nous faisons appel à un algorithme glouton dans lequel nous fixons les plus petites valeurs disponibles à chacune des différences qui interviennent dans le calcul de la borne inférieure en commençant toujours par les différences ayant le plus grand poids. Dans certaines situations, dépendamment des valeurs interdites, nous pouvons réduire considérablement les étapes nécessaires pour calculer cette borne. Pour cela, nous devons considérer trois

cas, soient si  $m \leq k - 1$ , ou si  $m \geq M - k + 1$  ou encore si  $k - 1 < m < M - k + 1$ .

Une première réduction simple est obtenue en conservant la somme de toutes les différences mesurées qui prend part dans le calcul de la borne inférieure dans la structure  $\text{PARTIAL-BOUND}(m)$ . Cette solution nous évite de recalculer la même somme à chaque itération. Elle est maintenue à jour par :

$$\text{PARTIAL-BOUND}(m) = \text{PARTIAL-BOUND}(m - 1) + w_{m,m-1}^k * \ell_{m,m-1} + \sum_{j=m-k}^{m-1} \ell_{m,j}.$$

*Preuve.* Soit  $m$  la nouvelle marque fixée. Nous devons ajouter à  $\text{PARTIAL-BOUND}(m)$  la nouvelle différence générée avec son poids respectif  $w_{m,m-1}^k$ . Par la suite nous devons considérer toutes les distances déduites de  $\text{LENGTH}(m) - \text{LENGTH}(m - i)$  pour  $i \in \{1, 2, \dots, k - 1\}$ .

Pour obtenir la borne inférieure, il suffit de compléter cette somme avec l'ensemble des plus petites valeurs disponibles pour les arêtes non fixées (qui n'ont pas encore de valeur numérique).

**Proposition 6** Soient  $m$  une marque que nous venons de fixer, avec  $m \leq k - 1$  et  $\mathcal{A}$  la somme des valeurs comprises entre 1 et  $kM - \frac{k(k+1)}{2}$  inclusivement. Soient  $\ell'_{ij} = \text{length}(i) - \text{length}(j)$  pour toutes les marques fixées et  $\ell_{M,M-1}$  la plus petite valeur disponible supérieure à  $\ell'_{21}$ . La borne inférieure est obtenue par :

$$\begin{aligned} \frac{k(k+1)}{2} \underline{S} = & \mathcal{A} + \text{PARTIAL-BOUND}(m) + \sum_{i=2}^m w_{i,i-1}^k (\ell_{M-i+2,M-i+1}) \\ & + \sum_{i=m+1}^k w_{i,i-1}^k (\ell_{i,i-1} + \ell_{M-i+2,M-i+1}) - \sum_{i=2}^m \sum_{j=1}^{i-1} \ell'_{ij}. \end{aligned}$$

*Preuve.* Notons  $\ell'_{ij} = \text{length}(i) - \text{length}(j)$ , la distance entre les marques fixées et  $\ell_{ij} = \{\text{plus petites valeurs disponibles}\}$ , l'équation (2.1) peut s'écrire sous la forme

suivante :

$$\begin{aligned} \frac{k(k+1)}{2} \underline{S} = & \sum_{i=2}^m \sum_{j=i-k}^{i-1} \ell'_{ij} + \sum_{i=2}^m w_{i,i-1}^k \ell'_{i,i-1} + \sum_{i=m+1}^M \sum_{j=i-k}^{i-1} \ell_{ij} \\ & + \sum_{i=2}^m w_{i,i-1}^k (\ell_{M-i+2,M-i+1}) + \sum_{i=m+1}^k w_{i,i-1}^k (\ell_{i,i-1} + \ell_{M-i+2,M-i+1}). \end{aligned}$$

Puisque la somme des différences générées à partir des marques fixées et qui interviennent dans la borne inférieure est gardée dans PARTIAL-BOUND( $m$ ), c'est-à-dire  $\sum_{i=2}^m \sum_{j=i-k}^{i-1} \ell'_{ij} + \sum_{i=2}^m w_{i,i-1}^k \ell'_{i,i-1}$ , nous avons :

$$\begin{aligned} \frac{k(k+1)}{2} \underline{S} = & \text{PARTIAL-BOUND}(m) + \sum_{i=m+1}^M \sum_{j=i-k}^{i-1} \ell_{ij} \\ & + \sum_{i=2}^m w_{i,i-1}^k (\ell_{M-i+2,M-i+1}) + \sum_{i=m+1}^k w_{i,i-1}^k (\ell_{i,i-1} + \ell_{M-i+2,M-i+1}). \end{aligned}$$

Cependant, puisque  $m \leq k-1$ , toutes les différences calculées à partir des marques fixées sont considérées dans la borne inférieure. De plus, nous savons qu'une règle de  $M$  marques génère  $kM$  différences qui sont dans la borne inférieure. Se basant sur le fait que les  $k$  premières marques génèrent  $k$  distances chacune, le nombre total de différences est donc  $kM - \frac{k(k+1)}{2}$ . Nous pouvons donc écrire :

$$\mathcal{A} = \sum_{\ell=1}^{kM - \frac{k(k+1)}{2}} a_{\ell} = \sum_{i=2}^m \sum_{j=i-k}^{i-1} \ell'_{ij} + \sum_{i=m+1}^M \sum_{j=i-k}^{i-1} \ell_{ij},$$

où les  $a_{\ell}$  sont égaux à un des  $\ell'_{ij}$ , pour  $\ell \leq \frac{m(m+1)}{2}$ , et  $a_{\frac{m(m+1)}{2} + i}$  à un des  $\ell_{ij}$ . En les substituant dans la formule de la borne inférieure, nous obtenons :

$$\mathcal{A} + \text{PARTIAL-BOUND}(m) - \sum_{i=2}^m \sum_{j=1}^{i-1} \ell'_{ij} = \sum_{i=2}^m \sum_{j=i-k}^{i-1} \ell'_{ij} + \sum_{i=2}^m w_{i,i-1}^k \ell'_{i,i-1} + \sum_{i=m+1}^M \sum_{j=i-k}^{i-1} \ell_{ij}.$$

Si nous savons en plus que toutes les valeurs des  $a_i$  sont inférieures à  $kM - \frac{k(k+1)}{2}$ , alors les valeurs comprises entre 1 et  $kM - \frac{k(k+1)}{2}$  seront nécessaires pour compléter

la borne inférieure et à partir de la formule  $\sum_{i=1}^n i = \frac{n(n+1)}{2}$ , nous obtenons :

$$\begin{aligned} \frac{k(k+1)}{2} \underline{S} = \mathcal{A} + \text{PARTIAL-BOUND}(m) + \sum_{i=2}^m w_{i,i-1}^k (\ell_{M-i+2,M-i+1}) \\ + \sum_{i=m+1}^k w_{i,i-1}^k (\ell_{i,i-1} + \ell_{M-i+2,M-i+1}) - \sum_{i=2}^m \sum_{j=1}^{i-1} \ell'_{ij}, \end{aligned}$$

où  $\mathcal{A} = \frac{(kM - \frac{k(k+1)}{2})(kM - \frac{k(k+1)}{2} + 1)}{2}$ . ■

**Proposition 7** Soient  $m \geq M - k + 1$  et  $\ell_{M,M-1}$  la plus petite valeur disponible supérieure à  $\ell_{2,1}$ , la borne inférieure s'écrit :

$$\begin{aligned} \frac{k(k+1)}{2} \underline{S} = \text{PARTIAL-BOUND}(m) + \sum_{i=2}^k w_{i,i-1}^k (\ell_{M-i+2,M-i+1}) \\ + \sum_{i=1}^{\frac{(M-m-1)(M-m-2)}{2}} \ell_{M-i+1,M-i}. \end{aligned}$$

*Preuve.* Si  $m \geq M - k + 1$ , la borne inférieure s'obtient en utilisant l'algorithme glouton mentionné précédemment. Nous devons chercher les  $\frac{(M-m-1)(M-m)}{2}$  plus petites valeurs disponibles  $(\ell_{ij})$  pour toutes les différences non considérées en commençant par celles avec le plus grand poids  $w_{ij}^k$  pour  $i \in \{1, 2, \dots, M - m - 1\}$  et pour  $j \in \{i - k, i - k + 1, \dots, i\}$ . La borne inférieure s'obtient en ajoutant ces valeurs à  $\text{PARTIAL-BOUND}(m)$ . Cela correspond à sommer tous les chemins considérés. Pour obtenir la borne inférieure, nous devons diviser cette somme par le nombre de chemins. ■

En ce qui concerne le dernier cas, c'est-à-dire  $k - 1 \leq m \leq M - k + 1$ , trouver un moyen pour réduire le nombre d'étapes risquerait de coûter aussi cher, sinon plus que de chercher systématiquement chacune de ces valeurs. Ceci veut donc dire

que nous devons trouver toutes les plus petites valeurs disponibles pour compléter les chemins.

**Proposition 8** Soient  $k - 1 \leq m \leq M - k + 1$  et  $\ell_{M-1, M-2}$  la plus petite valeur disponible supérieure à  $\ell_{1,0}$ , nous calculons la borne inférieure par :

$$\frac{k(k+1)}{2} \underline{S} = \text{PARTIAL-BOUND}(m) + \sum_{i=2}^k w_{i,i-1}^k(\ell_{M-i+2, M-i+1}) + \sum_{i=m+1}^{kM - \frac{(k-2)(k+1)}{2}} \sum_{j=1}^k \ell_{i,i-j}.$$

La complexité des calculs de la borne inférieure nous oblige à réduire le nombre de fois que nous utilisons ce test. Dans ce mémoire, nous ne nous intéressons qu'aux bornes inférieures pour les cas où  $m$  est inférieure à 4 et pour lesquels  $\ell_{2,1}$  et  $\ell_{3,2}$  sont supérieures à  $k - 1$ . Ces cas particuliers permettent de dériver des formules analytiques.

**Proposition 9** Soit  $b$  la position de la deuxième marque telle que  $b \geq k - 1$  et  $S^0$  la valeur obtenue par la proposition 5, la valeur de la borne inférieure s'obtient par :

$$\frac{k(k+1)}{2} \underline{S}^1 = \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - \frac{2}{3}k(k-1)(k-2).$$

*Preuve.* Avec la seconde marque fixée à  $b$ , c'est-à-dire  $\ell_{2,1} = b$ , ceci entraîne que  $\ell_{M, M-1} = b + 1$  (test de symétrie) puisque  $b + 1$  est la première valeur disponible supérieure à  $b$ . La nouvelle borne inférieure est mise à jour en remplaçant les valeurs 1 et 2 de  $\ell_{2,1}$  et  $\ell_{M, M-1}$  respectivement par  $b$  et  $b + 1$ . Cet ajustement doit être effectué pour toutes les valeurs que nous avons attribué pour les  $\ell_{3,2}$  à  $\ell_{k, k-1}$  et les  $\ell_{M-k+1, M-k}$  à  $\ell_{M, M-1}$ , puisque ces valeurs ont aussi changé. Ces modifications

conduisent à la formule suivante :

$$\begin{aligned}
 \frac{k(k+1)}{2} \underline{S^1} &= \frac{k(k+1)}{2} S^0 + \frac{k(k+1)}{2} ((b+b+1) - (1+2)) \\
 &\quad - 4 \sum_{i=2}^{k-1} \frac{(k-i+1)(k-i)}{2} \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - 2 \sum_{i=2}^{k-1} (k-i+1)(k-i) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - 2 \sum_{i=2}^{k-1} (i^2 - (2k+1)i + k(k+1)).
 \end{aligned}$$

Sachant que  $\sum_{i=1}^n i = \frac{n(n+1)}{2}$  and  $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$ , nous obtenons :

$$\begin{aligned}
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) \\
 &\quad - 2 \left( \frac{k(k-1)(2k-1)}{6} - 1 - (2k+1) \left( \frac{k(k-1)}{2} - 1 \right) + (k-2)(k+1) \right) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - 2 \left( \frac{k(k-1)(2k-1)}{6} - \frac{(k-2)(k+1)}{2} - 1 \right) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - 2 \left( \frac{k(k-1)(2k-1)}{6} - \frac{3(k-2)(k+1)}{6} - 1 \right) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - 2 \left( \frac{2k^3 - 6k^2 + 4k}{6} \right) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) - \frac{2}{3} (k(k-1)(k-2)).
 \end{aligned}$$

■

De même, en utilisant le même processus pour  $m = 3$  avec  $\ell_{2,1}$  et  $\ell_{3,2}$  supérieurs à  $k-1$ , nous obtenons la proposition suivante :

**Proposition 10** Soit  $b$  et  $c$  les positions de la deuxième et troisième marques telles que  $b$  et  $c$  sont supérieures à  $k-1$ . Soit  $S^0$  une borne inférieure calculée à partir de la proposition 5, la nouvelle valeur de la borne inférieure s'obtient par :

$$\frac{k(k+1)}{2} \underline{S^2} = \frac{k(k+1)}{2} S^0 + \frac{k(k-1)}{2} (b-1) + \frac{(k-1)(k-2)}{2} (c-6) - (k-1)(k-2)(k-3).$$

*Preuve.* En attribuant les valeurs  $b$  et  $c$  à  $\ell_{2,1}$  et  $\ell_{3,2}$ , les anciennes valeurs disponibles attribuées à  $\ell_{M,M-1}$  et  $\ell_{M-1,M-2}$  doivent être modifiées. De même pour chaque paire  $(\ell_{i,i-1}, \ell_{M-i+1,M-i})$ ,  $\text{LENGTH}(M-i)$  pour tout  $i = 3, 4, \dots, k-1$ , il y aura une réduction de 6 unité par rapport à l'affectation initiale. De sorte que la nouvelle formule devient :

$$\begin{aligned}
 \frac{k(k+1)}{2} \underline{S^2} &= \frac{k(k+1)}{2} S^0 + \frac{k(k-1)}{2} ((b+b+1) - (1+2)) \\
 &\quad + \frac{(k-1)(k-2)}{2} ((c+1) - 7) - 6 \sum_{i=3}^{k-1} \frac{(k-i+1)(k-i)}{2} \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
 &\quad - 3 \sum_{i=3}^{k-1} (i^2 - (2k+1)i + k(k+1)) \\
 &= \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6)
 \end{aligned}$$



$$\begin{aligned}
& -3 \left( \frac{k(k-1)(2n-1)}{6} - 5 - (2k+1) \left( \frac{k(k-1)}{2} - 3 \right) \right) \\
& -3((k-3)k(k+1)) \\
= & \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
& -3 \left( \frac{k(k-1)(2k-1)}{6} - \frac{(2k+1)(k+2)(k-3)}{2} \right) \\
& -3((k-3)k(k+1) - 5) \\
= & \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
& -3 \left( \frac{k(k-1)(2k-1)}{6} - \frac{(3k+2)(k-3)}{2} - 1 \right) \\
= & \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
& -3 \left( \frac{2k^3 - 12k^2 + 22k + 18}{6} - 5 \right) \\
= & \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
& -3 \left( \frac{k^3 - 12k^2 + 11k + 6}{3} \right) \\
= & \frac{k(k+1)}{2} S^0 + k(k-1)(b-1) + \frac{(k-1)(k-2)}{2} (c-6) \\
& -(k-1)(k-2)(k-3).
\end{aligned}$$

■

#### 2.5.3.4 Borne inférieure dans G-DFS2

Examinons les étapes nécessaires du calcul de la borne inférieure pour l'ordre 2. Nous considérons 2 cas, soient  $k-1 \leq m \leq M-k+1$  et  $m < k-1$  ou  $m > M-k+1$ . Pour le cas où  $m \leq k-1$  or  $m \geq M-k+1$  et  $\ell_{ij}$  inférieure à  $kM - \frac{k(k+1)}{2}$ , nous utilisons une proposition similaire à celle de la proposition 6 pour calculer la borne inférieure.

**Proposition 11** Soient  $m \leq k-1$  ou  $m \geq M-k+1$  et  $\ell_{ij}$  inférieure à  $kn - \frac{k(k+1)}{2}$  et  $A$  la somme des valeurs comprises entre 1 et  $kM - \frac{k(k+1)}{2}$  inclusivement, la borne

inférieure est :

$$\frac{k(k+1)}{2} \underline{S} = A + \sum_{i=1}^{k-1} \sum_{j=1}^i w_{ij}^k (\ell_{i,i-1} + \ell_{M-i,M-i-1}),$$

où  $\ell_{ij} = \text{LENGTH}(i) - \text{LENGTH}(j)$  si  $i$  est une marque fixée et  $\ell_{ij} = \{\text{plus petites valeurs disponibles}\}$  autrement.

*Preuve.* Lorsque  $m \leq k-1$  or  $m \geq M-k+1$  toutes les distances entre les paires de marques fixées sont considérées dans le calcul de la borne inférieure. Ainsi pour obtenir cette borne, nous devons compléter ces valeurs avec l'ensemble des plus petites valeurs disponibles. Ce qui correspond  $A$  auquel on ajoute les  $\ell_{ij}$  avec leur poids respectivement. ■

Cependant dans le cas où  $k-1 \leq m \leq M-k+1$ , nous ne pouvons faire autrement que de compléter  $\text{PARTIAL-BOUND}(m)$  avec les plus petites valeurs disponibles. Ceci conduit à la proposition suivante :

**Proposition 12** Soient  $m$  une marque nouvellement fixée telle que  $k-1 \leq m \leq M-k+1$  et  $\ell_{ij}$  les plus petites valeurs disponibles, alors la borne inférieure est :

$$\frac{k(k+1)}{2} \underline{S} = \text{PARTIAL-BOUND}(m) + \sum_{i=m+1}^{M-m-1} \sum_{j=1}^{k-1} \ell_{i,i-j}.$$

Dans la version améliorée G-DFS1 de GARSP et G-DFS2, le test de la borne inférieure sera limité au cas où les positions des 3 premières marques changent. Par conséquence les propositions 9 et 10 seront utilisées.

## 2.6 Résultats et analyse

Nous présentons ci-dessous les résultats obtenus suite à l'introduction des tests de réduction et de borne inférieure dans GARSP. Ces résultats ont été obtenus sur un Sun Ultra 1 avec 256 M de mémoire. Le tableau 2.1 indique les temps de dérivation des règles de Golomb optimales de l'algorithme original GARSP et des nouvelles versions améliorées. Le deuxième tableau (tableau 2.2) indique le nombre de noeuds visités lors de la recherche.

GARSP : Programme original

G-DFS1a : ordre 1 avec borne inférieure calculée après avoir fixé une marque.

G-DFS1b : ordre 1 avec borne inférieure calculée après avoir fixé deux marques.

G-DFS2a : ordre 2 avec borne inférieure calculée après avoir fixé une marque.

G-DFS2b : ordre 2 avec borne inférieure calculée après avoir fixé deux marques.

Tableau 2.1 – Temps de dérivation des règles de Golomb

M	Temps de dérivations (en secondes)				
	GARSP	G-DFS1a	G-DFS1b	G-DFS2a	G-DFS2b
4 - 10	$\leq 10^{-2}$				
11	1	$\leq 10^{-2}$	1	$\leq 10^{-2}$	$\leq 10^{-2}$
12	2	1	1	3	3
13	48	17	15	74	64
14	518	286	258	1 913	1 680
15	8 277	5 419	4 958	45 240	38 634

Tableau 2.2 – *Nombre de noeuds examinés*

M	Nombre de noeuds explorés				
	GARSP	G-DFS1a	G-DFS1b	G-DFS2a	G-DFS2b
4	10	4	4	4	3
5	46	24	24	10	9
6	162	59	59	16	14
7	953	234	234	27	24
8	3 294	1 040	1 040	51	46
9	11 595	2 740	2 740	1 267	1 258
10	32 943	7 871	7 871	615	606
11	993 343	316 307	264 062	297	286
12	$\approx 2.933 \times 10^6$	851 419	782 304	$\approx 2.897 \times 10^6$	$\approx 2.189 \times 10^6$
13	$\approx 8.895 \times 10^7$	$\approx 3.5 \times 10^7$	$\approx 3.143 \times 10^7$	$\approx 7.392 \times 10^7$	$\approx 7.1 \times 10^7$
14	$\approx 1.015 \times 10^9$	$\approx 6.117 \times 10^8$	$\approx 5.621 \times 10^8$	$\approx 1.935 \times 10^9$	$\approx 1.756 \times 10^9$
15	$\approx 1.850 \times 10^{10}$	$\approx 1.184 \times 10^{10}$	$\approx 1.120 \times 10^{10}$	$\approx 4.1 \times 10^{10}$	$\approx 3.789 \times 10^{10}$

Les résultats démontrent une amélioration substantielle lorsque les nouveaux tests de réductions ont été introduits dans GARSP. Ces améliorations sont de l'ordre de 40% à 70%. Cependant, au fur et à mesure que le nombre de marques augmente, les pourcentages de gain diminuent. Cette réduction s'explique par une augmentation exponentielle du domaine réalisable alors que la proportion réduite par les tests est de moins en moins importante. Observer que la borne inférieure n'est calculée qu'une fois que les valeurs sur les deux premières marques changent. Par conséquent le test de la borne inférieure a de moins en moins d'effet au fur et à mesure que  $M$  augmente.

D'autre part, l'ordre 2 n'est pas une stratégie intéressante. Nous constatons

qu'il coûte plus cher, malgré l'efficacité des nouveaux tests de symétrie et de borne inférieure dans ce dernier. Observer que, dans cette stratégie, nous déterminons les règles en calculant les deux segments qui les composent. Nous perdons donc la simplicité des opérations de mise à jour développées pour l'ordre 1. Les structures de bitmaps nécessitent plus d'opérations pour les mettre à jour, ce qui réduit considérablement l'efficacité de l'algorithme. C'est pourquoi cette stratégie de recherche ne donne pas les résultats espérés compte tenu de l'efficacité des tests de réduction.

Il est logique de penser à calculer la borne inférieure après chaque itération pour accélérer les recherches, cependant les expériences démontrent qu'il ne s'agit pas d'un choix judicieux, car les gains en temps ne justifient pas les coûts de calcul des bornes.

En seconde partie, nous présentons un tableau des bornes supérieures obtenues par différents algorithmes de dérivation.

Tableau 2.3 – *Bornes supérieures*

M	Bornes supérieures				
	Plans	Glouton1 (sans borne)	Algo1 (avec borne)	Glouton2 (sans borne)	Algo2 (avec borne)
9	44	65	65	50	50
10	55	80	80	69	69
11	72	96	96	92	92
12	85	122	122	116	116
13	106	147	147	141	141
14	127	181	181	172	172
15	151	203	203	185	185
20	283	474	397	432	396
21	333	564	432	471	435
25	480	821	—	734	—
30	680	1 311	—	1 163	—
35	987	1 895	—	1 899	—
40	1 282	2 779	—	2 752	—
50	2 094	5 332	—	4 933	—
60	3 019	9 357	—	7 837	—
70	4 217	13 691	—	12 256	—
80	5 563	21 354	—	18 300	—
90	6 067	29 739	—	26 084	—
100	8 831	41 280	—	34 749	—

La première colonne indique la longueur des meilleures règles actuellement connues (Shearer [33]). Ces valeurs ont été obtenues par des algorithmes de constructions de plans projectifs et semi-affines. Les quatres dernières colonnes correspondent aux résultats obtenus par les algorithmes gloutons. Glouton1 et Glouton2

sont respectivement G-DFS1 et G-DFS2 sans processus d'optimisation. C'est-à-dire que les marques sont fixées les unes après les autres jusqu'à la dernière et la règle ainsi trouvée est celle fournie au tableau 2.3. La seule différence avec G-DFS1 et G-DFS2 est qu'aucun retour-arrière n'est effectué pour essayer de trouver une meilleure règle et aucun test de réduction n'est introduit dans Glouton1 et Glouton2. Nous appelons Algo1 et Algo2, les mêmes algorithmes d'énumération implicite (Glouton1 et Glouton2) dans lesquels la borne supérieure est fixée à  $M^2$  et dans lesquels les tests de bornes inférieures et de réduction sont introduits. Les résultats obtenus par les algorithmes gloutons correspondent à la longueur de la première règle valide trouvée.

Les résultats nous démontrent que les solutions fournies par les algorithmes de construction de plans projectifs et semi-affines sont nettement meilleures que celles fournies par les algorithmes gloutons. Cependant, il faut remarquer que ces résultats correspondent aux meilleures solutions connues. Par conséquence, une certaine optimisation a été effectuée pour calculer ces règles. Cependant, aucune donnée ne nous a été fournie sur les temps de calculs ou le nombre d'itérations nécessaire pour obtenir ces résultats. Il n'est donc pas possible de faire une comparaison juste entre les valeurs obtenues. Néanmoins nous pouvons constater à quel point les résultats fournis par les algorithmes gloutons sont loin de la solution optimale.

Nous constatons aussi que les bornes supérieures obtenues pour l'ordre 2 sont meilleures que celles fournies par l'ordre 1. Observer qu'à partir du moment où une certaine optimisation est nécessaire, la taille des règles que nous pouvons trouver dans un délai raisonnable est très limitée. Dans les deux algorithmes gloutons avec bornes, le simple ajout d'une borne supérieure nous empêche de calculer une règle valide avec un temps de calcul inférieur à une semaine sur un Sun Ultra 1 avec 256 M de mémoire pour les règles de plus de 25 marques.

Pour fin de comparaison, nous avons récupéré SEMIAFF.CC [37] un algorithme qui permet de construire les plans semi-affines [38] en utilisant les équations de récurrences [39] et à partir lesquels on déduit les règles de Golomb. Le programme est utilisé pour construire les quelques premières règles de Golomb pour un nombre premier de marques. Ces règles sont obtenues sans qu'il y est eu d'optimisation. Les résultats se trouvent dans le tableau ci-dessous.

Tableau 2.4 – *Bornes supérieures pour un nombre premier de marques*

M	Bornes supérieures				
	Plans	Glouton1 (sans borne)	Algo1 (avec borne)	Glouton2 (sans borne)	Algo2 (avec borne)
13	131	147	147	141	141
17	238	289	289	256	256
19	293	400	351	345	345
23	388	661	—	570	—
29	740	1 158	—	1 098	—
37	1 206	2 253	—	2 196	—
41	1 549	2 924	—	3 002	—
43	1 659	3 865	—	3 340	—
47	1 986	4 540	—	4 437	—
53	2 570	6 653	—	5 621	—
59	3 249	8 901	—	7 786	—
61	3 475	9 640	—	8 569	—
67	4 133	12 384	—	10 488	—
71	4 765	13 821	—	12 328	—
73	4 950	14 864	—	13 758	—
79	5 803	20 530	—	16 984	—
83	6 414	23 935	—	20 178	—
89	7 528	28 686	—	24 300	—
97	8 838	38 698	—	32 355	—



De nouveau, les résultats confirment que les règles obtenues par construction de plans sont de meilleure qualité. Cependant il faut remarquer qu'il n'est pas possible de vérifier leur optimalité, pour cela il faut construire tous les plans, ce qui n'est pas très pratique.

## 2.7 Conclusion

Les nouveaux tests de réduction ont permis d'améliorer substantiellement les temps de calcul de dérivation des règles de Golomb optimales. Cependant, malgré une réduction de 40%, les temps de calcul pour les règles de Golomb optimales avec un grand nombre de marques demeurent extrêmement longs. En ce qui concerne les tests de réduction et de bornes inférieures, il est peu probable que nous puissions améliorer davantage les algorithmes par cette stratégie. Observer que plus la borne est précise plus elle est chère à calculer. De sorte que, même si nous arrivons à calculer une borne encore plus précise, le coût en temps de calcul ne justifierait probablement pas leur utilisation. Il serait donc plus judicieux de développer des méthodes efficaces pour les calculer rapidement. Ces méthodes peuvent être sous forme d'optimisation des codes ou simplement par un choix judicieux des structures de données à utiliser.

En ce qui concerne la méthode de dérivation, les améliorations peuvent se faire à deux niveaux, soit dans la dérivation de la règle de Golomb elle-même et dans la partie du test de l'optimalité. D'abord dans le premier cas, de meilleures bornes supérieures pourront être introduites pour limiter les recherches. Ces valeurs peuvent être obtenues heuristiquement et par la suite introduites dans l'algorithme exacte. D'autre part dans le deuxième cas, bien qu'on puisse toujours développer de meilleurs tests de réduction, il serait peut-être bon de considérer une implantation en parallèle.

Nous savons qu'en général, une grande partie du temps de calcul est consacrée à démontrer que la règle trouvée est optimale. Pour améliorer davantage nous devons probablement développer de nouvelles approches de dérivation, peut-être par construction des plans. À première vue, ces algorithmes donnent de très bonnes

bornes par rapport aux algorithmes gloutons, ce qui permettrait probablement d'améliorer l'efficacité des tests de la borne inférieure. Cependant le gros problème reste à démontrer que la règle est optimale, puisqu'il n'est pas question de générer tous les plans.

Une autre alternative serait d'utiliser des métaheuristiques. Peu de travaux ont été effectués dans cette direction. Seul un algorithme génétique a été proposé par Soliday, Homaifar et Lebby[27] en 1995. Les métaheuristiques pourraient permettre d'obtenir d'assez bonnes règles (non optimales) dans un délai raisonnable, c'est-à-dire en quelques minutes alors que la solution optimale se calcule en années par une méthode exhaustive.

Une stratégie non explorée encore consiste à fusionner les méthodes heuristiques avec les méthodes exactes, c'est-à-dire dériver les règles pré-optimales à partir des méthodes heuristiques qu'on améliorerait et éventuellement une confirmation de l'optimalité de leur solution.

## CHAPITRE 3

# INTERMODULATION ET AFFECTATION DE FRÉQUENCES

### 3.1 Introduction

Ce chapitre est consacré à l'application des règles de Golomb aux problèmes d'affectation de fréquences avec élimination des produits de l'intermodulation. Ce problème apparaît dans celui de trouver une affectation satisfaisante dans un réseau de téléphonie cellulaire du type AMPS ou GSM.

Le problème est d'une importance primordiale, car ces dernières années ont connu une grande croissance dans ce domaine. Les demandes ne cessent d'augmenter et cela ne semble pas s'arrêter. Pour satisfaire les demandes, les compagnies doivent attribuer judicieusement la bande de fréquences qui sont à leur disposition. Leur objectif est de satisfaire la plus grande demande possible tout en fournissant un service de qualité acceptable, c'est-à-dire minimiser les problèmes d'interférences et la non-disponibilité des lignes.

Les travaux dans ce domaine sont peu nombreux, un des premiers a été entrepris par Babcock [2] qui portait sur les systèmes radios. Babcock fournissait des séquences d'attribution qui sont absentes de tous les produits d'intermodulation du 3<sup>e</sup> ordre. Dans un travail similaire, Fang et Sandrin [10] démontrent que trouver une affectation de fréquences satisfaisant les produits d'intermodulation est équivalent à trouver une règle de Golomb. Nous reprenons cette preuve dans la section 3.4.

Nous proposons dans ce chapitre d'étudier les contraintes liées à l'intermodulation pour les problèmes d'affectation de fréquences. Une affectation satisfaisante

doit respecter les contraintes liées aux interférences co-canal, de canaux adjacents et d'intermodulation du 3<sup>e</sup> ordre (2 et 3 canaux). Nous proposons par la suite, un algorithme de recherche qui vérifie si une solution réalisable existe.

Ce chapitre est divisé en 7 sections. D'abord, dans la section 3.2, nous définissons quelques termes se rapportant à la téléphonie cellulaire. Ensuite nous exposons le problème à résoudre dans la section 3.3, qui sera suivi dans la section 3.4 d'une comparaison des règles de Golomb avec le problème d'affectation de fréquences. Les sections 3.5 et 3.6 portent sur la description de l'algorithme et de la borne inférieure. Finalement, nous terminons avec une présentation des résultats dans la section 3.7.

## 3.2 Définition

Un réseau cellulaire est une représentation graphique d'une région géographique à l'intérieur de laquelle on désire desservir un ensemble de clients. Le réseau est subdivisé en un ensemble de **sites** ; les sites seront notés  $S_q$  de façon générique. Chaque site est, à son tour, subdivisé en **cellules** ou **secteurs**. De manière identique, ces cellules seront notées  $C_i$ . Généralement on retrouve 2 à 3 cellules par site et une cellule comporte en moyenne 2 antennes. Dans le cadre de ce mémoire, nous considérons qu'une cellule ne comporte qu'une seule antenne avec une certaine demande en canaux. Ceci n'est pas une grande restriction, puisque si cela n'était pas le cas, il suffit de subdiviser la cellule et répartir les demandes de manière à obtenir une seule antenne par cellule. Cette hypothèse est très courante dans la littérature. Le problème consiste donc à attribuer une fréquence à chacun des canaux.

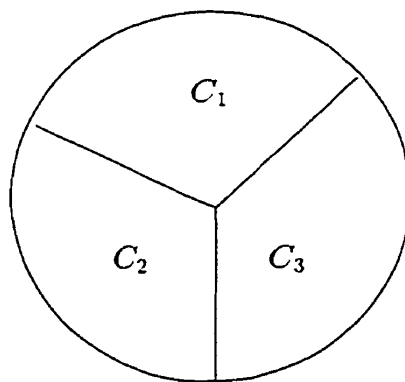


Figure 3.1 – Site

Selon la technologie utilisée, un certain nombre de fréquences est attribué à chacune des antennes. Par exemple, dans un grand réseau urbain, comme ceux de Bell Mobilité, il peut y avoir jusqu'à 15 canaux en FDMA ("Frequency Division Multiple Access"). Lorsque la technologie GSM ("Global System for Mobile" voir

Lee[21] pour plus de détails) est utilisée, la demande moyenne en canaux par cellule est de 3. Ceci s'explique par le fait qu'un canal FDMA est de 30 kHz alors qu'en GSM il est de 200 kHz. Étant donné que la largeur de bande des canaux est en moyenne 5 fois plus grande en GSM qu'en FDMA, le réseau GSM n'a pas besoin d'autant de canaux qu'en FDMA pour maintenir le même niveau de services.

La largeur de bande, c'est-à-dire la différence entre la plus haute fréquence et la plus basse fréquence disponibles sera subdivisée en plusieurs bandes de même largeur. Chacune de ses bandes reçoivent un numéro. Elles sont notées par  $f_k$  et  $f_l$  et l'ensemble de toutes les bandes disponibles est noté  $F$ . L'affectation des fréquences aux canaux se fera en attribuant des numéros aux cellules. Les fréquences seront choisies de manière à respecter les différentes contraintes d'interférences. Nous considérons les quatres suivantes :

**Interférences co-canaux** : une même fréquence peut être attribuée à deux cellules différentes  $C_i$  et  $C_j$  si elles sont complètement compatibles, c'est-à-dire s'il n'y a aucune possibilité d'interférences entre les fréquences de ces deux cellules. Nous représentons cette compatibilité par une matrice symétrique, notée  $M^{co}$ , dont chaque élément  $m_{ij}^{co}$  prend une valeur entre 0 et 10. La valeur 0 correspond à une compatibilité parfaite et 10 à une incompatibilité totale. Généralement on fixe un seuil d'acceptabilité, notée  $m^{co}$ , au-dessus duquel les cellules  $C_i$  et  $C_j$  ne peuvent pas avoir les mêmes fréquences.

**Interférences canaux adjacents** : deux cellules  $C_i$  et  $C_j$  peuvent utiliser des fréquences adjacentes si elles sont parfaitement compatibles. Nous représentons cette mesure de compatibilité entre les cellules à l'aide d'une matrice symétrique, notée  $M^{adj}$ . Chaque élément  $m_{ij}^{adj}$  de la matrice prendra des valeurs comprises entre 0 et 10. Ces valeurs indiquent le degré d'incompatibilité entre les cellules avec 0 comme

parfaitement compatible et 10 totalement incompatible. De même, on définit un seuil d'acceptabilité noté  $m^{adj}$  au-dessus duquel les cellules  $C_i$  et  $C_j$  ne pourront pas recevoir des fréquences adjacentes.

**Contraintes d'espacement sur les antennes :** pour être en mesure de fonctionner correctement, une certaine distance  $esp_i$  de séparation doit exister entre les fréquences  $f_k$  et  $f_l$  fixées sur une même antenne d'une cellule  $C_i$  donnée.

**Intermodulation du 3<sup>e</sup> ordre :** les produits de l'intermodulation proviennent de la somme des fréquences fixées à chacune des cellules. Les combinaisons de ces fréquences génèrent des canaux qui peuvent entrer en interférence avec les fréquences des autres cellules. Ainsi le produit de l'intermodulation des canaux d'une cellule  $C_i$  doit être évité dans la cellule  $C_j$  si nous décidons de traiter l'intermodulation entre ces deux cellules. Une matrice  $M^{mod}$ , de 1 ou 0, sera utilisé pour indiquer s'il faut traiter ou non les produits de l'intermodulation entre les cellules. Ces éléments sont notés  $m_{ij}^{mod}$ .

Cette matrice n'est pas nécessairement symétrique, mais dans le cadre de cette étude nous supposons qu'elle est. Nous détaillons davantage ces contraintes dans la section qui suit.



### 3.3 Affectation de fréquences avec élimination de l'intermodulation du 3<sup>e</sup> ordre

Nous formalisons dans cette section les quatre contraintes d'interférences considérées dans le problème d'affectation de fréquences.

#### 3.3.1 Interférences co-canal

Pour satisfaire la grande demande en téléphonie cellulaire et augmenter l'efficacité d'utilisation des fréquences qui sont à notre disposition, certaines fréquences doivent être réutilisées dans plusieurs cellules. Cependant, si cette réaffectation se faisait de manière aléatoire, elle pourrait causer d'importantes interférences qui dégraderaient considérablement la qualité du service. Le critère de réutilisation est basé sur une distance minimale de séparation entre les cellules, c'est-à-dire qu'une même fréquence peut être attribuée dans deux cellules différentes si elles sont séparées d'une distance minimale. Cette distance dépend de la distance géographique et des conditions de propagation.

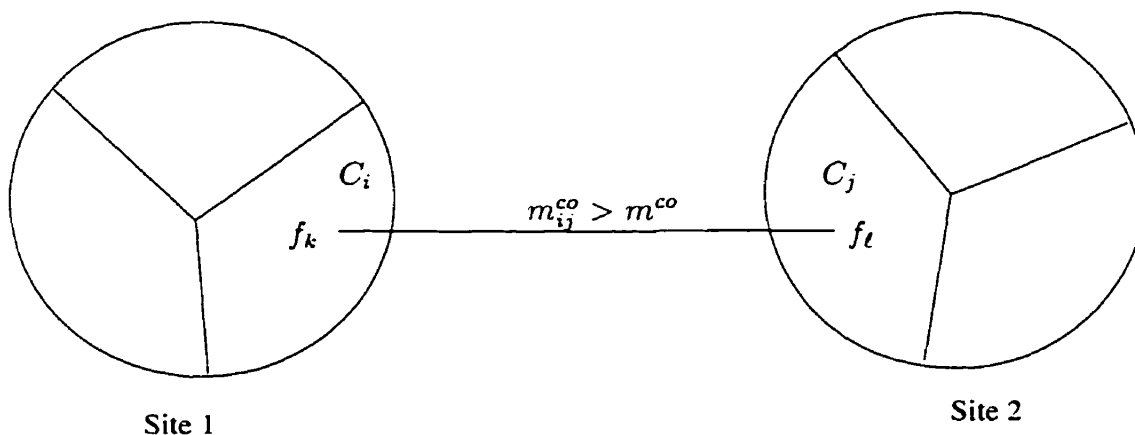


Figure 3.2 – Interférences co-canal

**Contrainte 1.** Soient  $f_k$  affectée à  $C_i$  et  $f_l$  affectée à  $C_j$ . Si  $m_{ij}^{co} > m^{co}$ , alors  $|f_k - f_l| \geq 1$ .

### 3.3.2 Interférences de canaux adjacents

L'interférence causée par les canaux adjacents se produit lorsque les signaux adjacents des cellules voisines sont reçus avec une puissance trop importante à l'intérieur d'une cellule de sorte qu'elle détériore la qualité de réception de ses propres signaux.

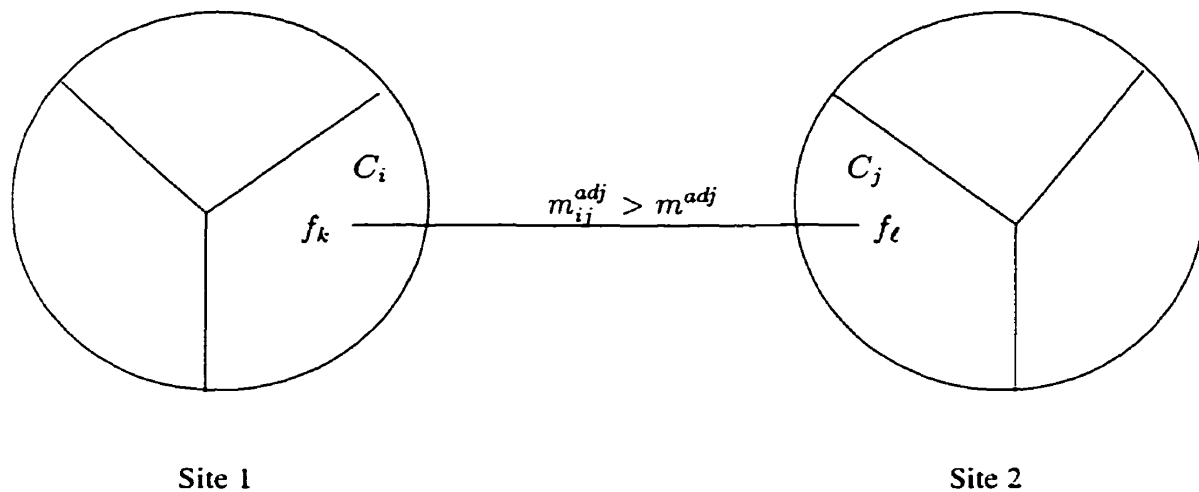


Figure 3.3 – Interférences de canaux adjacents

**Contrainte 2.** Soient  $f_k$  affectée à  $C_i$  et  $f_l$  affectée à  $C_j$ . Si  $m_{ij}^{adj} > m^{adj}$ , alors  $|f_k - f_l| \geq 2$ .

### 3.3.3 Contraintes d'espacement sur une antenne

Dans une cellule, les fréquences affectées à une même antenne doivent être séparées d'un espacement minimal de fonctionnement. Dans le cadre des expériences réalisées, nous avons fixé cette valeur à 17 pour les réseaux FDMA.

**Contrainte 3.** Si  $f_k$  et  $f_l$  affectées à  $C_i$ , alors  $|f_k - f_l| \geq 17$ .

### 3.3.4 Intermodulation du 3<sup>e</sup> ordre

Pour les produits d'intermodulation du 3<sup>e</sup> ordre, nous devons considérer ceux provenant de 2 canaux et de 3 canaux. Ces produits sont détaillés ci-dessous.

#### 3.3.4.1 Deux canaux

**Cas 1a.** Soient  $f_{k_1}, f_{k_2}$  affectées à  $C_i$  et  $f_\ell$  affectée à  $C_j$ , où  $C_i$  et  $C_j$  appartiennent à des sites différents. Si  $m_{ij}^{mod} = 1$ , alors

$$2f_{k_1} - f_{k_2} \neq f_\ell$$

$$2f_{k_2} - f_{k_1} \neq f_\ell.$$

**Cas 1b.** Soient  $f_{k_1}$  affectée à  $C_{i_1}$ ,  $f_{k_2}$  affectée à  $C_{i_2}$  et  $f_\ell$  affectée à  $C_j$ , où  $C_{i_1}$  et  $C_{i_2}$  appartiennent au même site et  $C_j$  à un site différent. Si  $m_{i_1j}^{mod} = m_{i_2j}^{mod} = 1$ , alors

$$2f_{k_1} - f_{k_2} \neq f_\ell$$

$$2f_{k_2} - f_{k_1} \neq f_\ell.$$

#### 3.3.4.2 Trois canaux

**Cas 2a.** Soient  $f_{k_1}, f_{k_2}, f_{k_3}$  affectées à  $C_i$  et  $f_\ell$  affectée à  $C_j$ , où  $C_i$  et  $C_j$  appartiennent à des sites différents. Si  $m_{ij}^{mod} = 1$ , alors

$$f_{k_1} + f_{k_2} - f_{k_3} \neq f_\ell$$

$$f_{k_1} + f_{k_3} - f_{k_2} \neq f_\ell$$

$$f_{k_2} + f_{k_3} - f_{k_1} \neq f_\ell.$$

**Cas 2b.** Soient  $f_{k_1}, f_{k_2}$  affectées à  $C_{i_1}$ ,  $f_{k_3}$  affectée à  $C_{i_2}$  et  $f_\ell$  affectée à  $C_j$ , où  $C_{i_1}$  et  $C_{i_2}$  appartiennent au même site et  $C_j$  à un site différent. Si  $m_{i_1j}^{mod} = m_{i_2j}^{mod} = 1$ , alors

$$f_{k_1} + f_{k_2} - f_{k_3} \neq f_\ell$$

$$f_{k_1} + f_{k_3} - f_{k_2} \neq f_\ell$$

$$f_{k_2} + f_{k_3} - f_{k_1} \neq f_\ell.$$

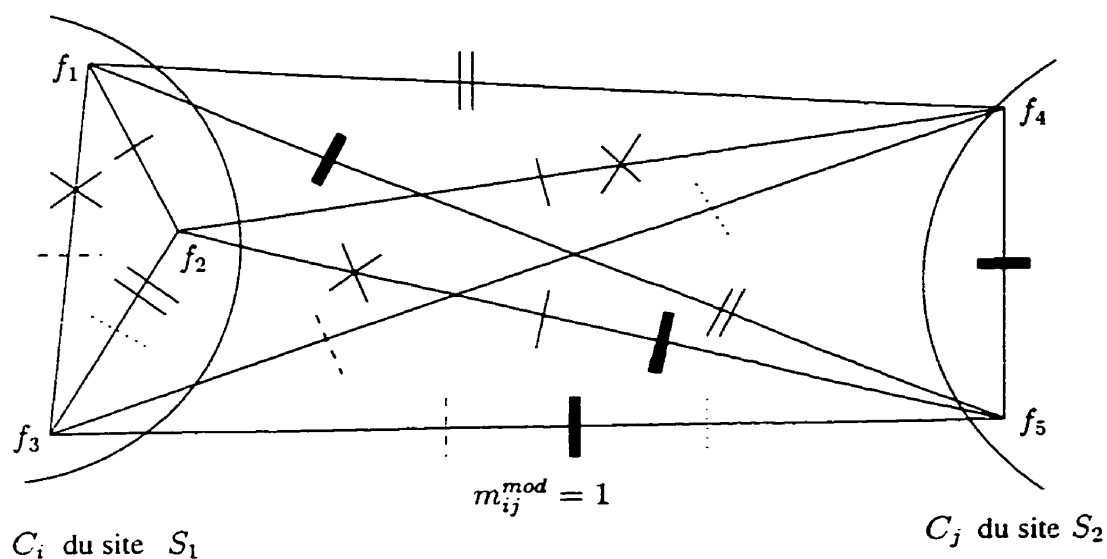
**Cas 2c.** Soient  $f_{k_1}$  affectée à  $C_{i_1}$ ,  $f_{k_2}$  affectée à  $C_{i_2}$ ,  $f_{k_3}$  affectée à  $C_{i_3}$  et  $f_\ell$  affectée à  $C_j$ , où  $C_{i_1}$ ,  $C_{i_2}$  et  $C_{i_3}$  appartiennent au même site et  $C_j$  à un site différent. Si  $m_{i_1j}^{mod} = m_{i_2j}^{mod} = m_{i_3j}^{mod} = 1$ , alors

$$f_{k_1} + f_{k_2} - f_{k_3} \neq f_\ell$$

$$f_{k_1} + f_{k_3} - f_{k_2} \neq f_\ell$$

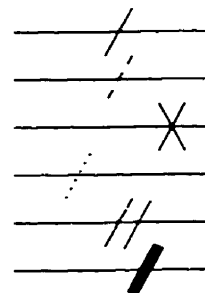
$$f_{k_2} + f_{k_3} - f_{k_1} \neq f_\ell.$$

Nous présentons ci-dessous un exemple simple illustrant les différents produits d'intermodulation considérés dans le problème d'affectation de fréquences. Soit 2 cellules  $C_i$  et  $C_j$  telles que  $m_{ij}^{mod} = m_{ji}^{mod} = 1$ , c'est-à-dire pour lesquelles nous devons considérer les interférences d'intermodulation du 3<sup>e</sup> ordre. La cellule  $C_i$  contient 3 fréquences, soient  $f_1, f_2, f_3$  et  $C_j$  en contient 2, soient  $f_4$  et  $f_5$ . Sans perte de généralité, supposons que  $f_1 < f_2 < f_3$  et  $f_4 < f_5$ . Observons que les produits d'intermodulation imposent une affectation telle que les différences entre toutes les paires de fréquences à l'intérieur d'une cellule soient distinctes des différences entre 2 fréquences prises dans des cellules différentes. Ces produits sont illustrés sur la figure 3.4.

Figure 3.4 - Intermodulation du 3<sup>e</sup> ordre

En appliquant les formules des produits d'intermodulation, nous obtenons des ensembles dans lesquels les différences doivent être toutes distinctes. Ces ensembles sont :

$$\begin{aligned}
 f_2 - f_1 &\neq f_4 - f_2 \neq f_5 - f_2 \\
 f_3 - f_1 &\neq f_4 - f_3 \neq f_5 - f_3 \\
 f_3 - f_1 &\neq f_4 - f_2 \neq f_5 - f_2 \\
 f_3 - f_2 &\neq f_4 - f_3 \neq f_5 - f_3 \\
 f_3 - f_2 &\neq f_4 - f_1 \neq f_5 - f_1 \\
 f_5 - f_4 &\neq f_3 - f_5 \neq f_2 - f_5 \neq f_1 - f_5.
 \end{aligned}$$



### 3.4 Règle de Golomb et problème d'intermodulation

Le problème d'affectation de fréquences sous contraintes reliées aux produits de l'intermodulation du 3<sup>e</sup> ordre dans un réseau cellulaire a été étudié par un nombre limité d'auteurs. Ces derniers sont arrivés à la conclusion qu'il s'agissait d'un problème équivalent à celui des ensembles disjoints de différences distinctes (noté DDD), où chacun de ces ensembles est en lui-même une règle de Golomb. Fang et Sandrin [10] ont démontré, dans un article apparu en 1977, que ce problème et celui de trouver des règles de Golomb étaient équivalents dans la mesure où il est nécessaire de considérer les produits de l'intermodulation intra-cellulaire. Cependant, compte tenu de la technologie actuelle, cela n'est plus le cas. Par conséquent, il n'est plus juste d'affirmer qu'ils sont équivalents. Les règles de Golomb sont plus contraignantes. Considérons le réseau réduit suivant de deux sites, noté  $S_1$  et  $S_2$ .

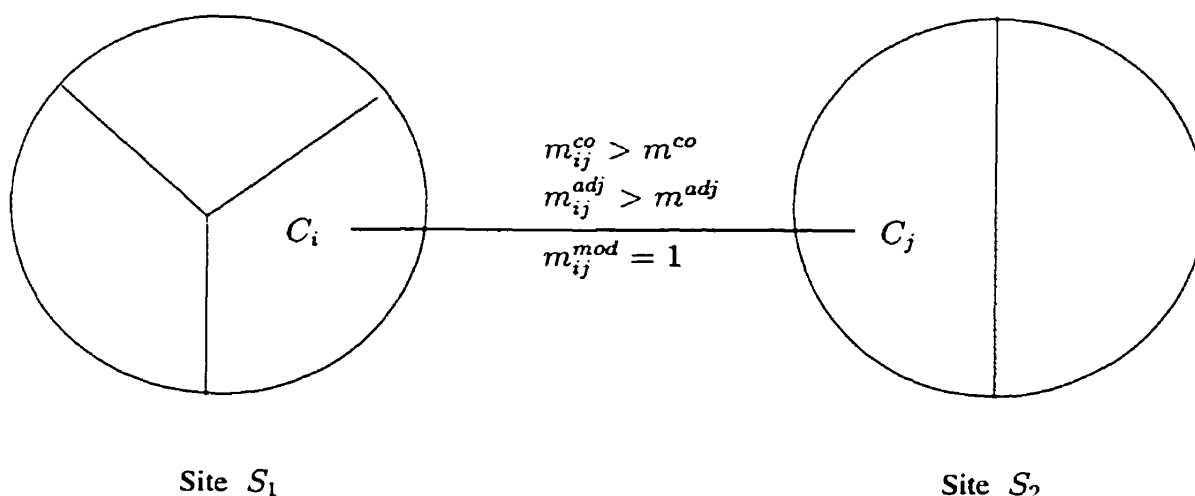


Figure 3.5 – Réseau à 2 sites

Les sites  $S_1$  et  $S_2$  sont subdivisés respectivement en 3 et 2 cellules comportant chacune une seule antenne. Chaque cellule nécessite une et une seule fréquence

pour couvrir sa région. Considérons aussi que toutes les contraintes d'interférences doivent être respectées, c'est-à-dire  $m_{ij}^{co} > m^{co}$ ,  $m_{ij}^{adj} > m^{adj}$ ,  $m_{ij}^{mod} = 1$  et  $esp_i = 17$ . Supposons que les fréquences  $f_1, f_2, f_3$  sont attribuées aux cellules du site  $S_1$  et  $f_4, f_5$  aux cellules du site  $S_2$ . Nous déduisons à partir des interférences co-canaux et canaux adjacents que  $f_1 \neq f_2 \neq f_3 \neq f_4 \neq f_5$ , cependant rien sur les différences entre les fréquences.

Soit  $f_i, f_j, f_k$  et  $f_s$  quatre fréquences quelconques. À partir des produits de l'intermodulation du 3<sup>me</sup> ordre, nous déduisons les inégalités suivantes :

1) 3<sup>me</sup> ordre 2 signaux

$$2f_i - f_j \neq f_s$$

2) 3<sup>me</sup> ordre 3 signaux

$$f_i + f_j - f_k \neq f_s.$$

Supposons que  $f_1, f_2, f_3, f_4$  et  $f_5$  sont les positions des marques d'une règle quelconque. Pour que l'ensemble forme une règle de Golomb à 5 marques, il faudrait que  $f_i - f_j \neq f_k - f_\ell$  pour tout  $i \neq j$  et  $k \neq \ell$ . Par conséquence si  $i = \ell$  ou  $j = k$ , nous obtenons les produits de l'intermodulation du 3<sup>e</sup> ordre avec deux canaux ( $2f_i - f_j \neq f_k$  ou  $2f_k - f_\ell \neq f_i$ ) et si toutes les quatres fréquences sont distinctes, nous retrouvons les produits de l'intermodulation du 3<sup>e</sup> ordre avec trois canaux. Ainsi, en considérant toutes les combinaisons possibles des fréquences, nous constatons que le problème d'affectation de fréquences et celui des règles de Golomb sont équivalents.

Cependant les technologies actuelles sont telles que les communications à l'intérieur d'une même cellule sont suffisamment bonnes, c'est-à-dire les signaux sont reçus avec une puissance assez importante pour qu'il ne soit plus nécessaire de

considérer les intermodulations intra-cellulaires. Ainsi, certaines contraintes seront omises. Le problème d'affectation des fréquences est donc moins contraignant que celui des règles de Golomb.

Par exemple, en appliquant les formules des produits de l'intermodulation du 3<sup>e</sup> ordre avec 2 canaux, nous obtenons des produits qui peuvent être regroupés pour former deux ensembles de différences :

$$2f_2 - f_1 \quad 2f_3 - f_2 \quad 2f_4 - f_3 \quad 2f_5 - f_4$$

$$2f_3 - f_1 \quad 2f_4 - f_2 \quad 2f_5 - f_3$$

$$2f_4 - f_1 \quad 2f_5 - f_2$$

$$2f_5 - f_1$$

$$2f_1 - f_2 \quad 2f_2 - f_3 \quad 2f_3 - f_{ell_4} \quad 2f_4 - f_5$$

$$2f_1 - f_3 \quad 2f_2 - f_4 \quad 2f_3 - f_5$$

$$2f_1 - f_4 \quad 2f_2 - f_5$$

$$2f_1 - f_5.$$

De même qu'en utilisant les formules avec 3 canaux, nous obtenons les produits suivants :

$$f_1 + f_2 - f_3 \quad f_1 + f_3 - f_2 \quad f_1 + f_4 - f_2 \quad f_1 + f_5 - f_2$$

$$f_1 + f_2 - f_4 \quad f_1 + f_3 - f_4 \quad f_1 + f_4 - f_3 \quad f_1 + f_5 - f_3$$

$$f_1 + f_2 - f_5 \quad f_1 + f_3 - f_5 \quad f_1 + f_4 - f_5 \quad f_1 + f_5 - f_4$$

$$f_2 + f_3 - f_1 \quad f_2 + f_4 - f_1 \quad f_2 + f_5 - f_1$$



$$f_2 + f_3 - f_4 \quad f_2 + f_4 - f_2 \quad f_2 + f_5 - f_3$$

$$f_2 + f_3 - f_5 \quad f_2 + f_4 - f_3 \quad f_2 + f_5 - f_4$$

$$f_3 + f_4 - f_1 \quad f_3 + f_5 - f_1$$

$$f_3 + f_4 - f_2 \quad f_3 + f_5 - f_2$$

$$f_3 + f_4 - f_5 \quad f_3 + f_5 - f_4$$

$$f_4 + f_5 - f_1$$

$$f_4 + f_5 - f_2$$

$$f_4 + f_5 - f_3.$$

Nous constatons qu'il y a énormément de contraintes à considérer, c'est-à-dire que les 5 fréquences choisies doivent être distinctes et différentes de chacun des produits. Cependant rien oblige que les différents produits d'intermodulation doivent être tous uniques. En effet en considérant  $f_1 = 1$ ,  $f_2 = 18$ ,  $f_3 = 36$ ,  $f_4 = 20$  et  $f_5 = 37$ , nous obtenons une affectation satisfaisant toutes les contraintes sans pour autant correspondre à une règle de Golomb, puisque la valeur 17 apparaît deux fois dans le triangle des différences.

$$\underline{1 \quad 18 \quad 20 \quad 36 \quad 37}$$

$$17 \quad 2 \quad 16 \quad 1$$

$$19 \quad 18 \quad 17$$

$$35 \quad 19$$

$$36$$

Triangle des différences.

Observer que les contraintes d'interférences reliées aux produits d'intermodulation sont tous respectées :

$$f_2 - f_1 \neq f_4 - f_2 \neq f_5 - f_2 \rightarrow 17 \neq 2 \neq 19$$

$$f_3 - f_1 \neq f_4 - f_3 \neq f_5 - f_3 \rightarrow 35 \neq -16 \neq 1$$

$$f_3 - f_1 \neq f_4 - f_2 \neq f_5 - f_2 \rightarrow 35 \neq 2 \neq 19$$

$$f_3 - f_2 \neq f_4 - f_3 \neq f_5 - f_3 \rightarrow 18 \neq -16 \neq 1$$

$$f_3 - f_2 \neq f_4 - f_1 \neq f_5 - f_1 \rightarrow 18 \neq 19 \neq 36$$

$$f_5 - f_4 \neq f_3 - f_5 \neq f_2 - f_5 \neq f_1 - f_5 \rightarrow 17 \neq -1 \neq -19 \neq -36$$

### 3.5 Algorithme de recherche

Nous proposons, dans cette section une méthode de recherche exhaustive en profondeur d'abord. Le principe consiste tout simplement à tour de rôle de choisir une cellule et lui attribuer une fréquence à un de ces canaux. Si la séquence d'attribution formée par l'ensemble des fréquences déjà affectées à laquelle s'ajoute la nouvelle respecte toutes les contraintes, nous procédons alors à la prochaine affectation.

À chaque itération, la cellule à traiter est choisie en considérant le niveau potentiel de difficulté à lui attribuer une affectation valide. Ce degré de difficulté s'évalue en calculant le nombre de fréquences qui sont interdites pour la cellule. Cette valeur est composée de deux parties, la première provient des fréquences déjà attribuées et la deuxième qui est une estimation du nombre de fréquences interdites qui seront générées lors des affectations futures. C'est-à-dire qu'il est possible de déterminer le nombre de fréquences additionnelles qui seront interdites en considérant les 4 contraintes d'interférences à respecter contenu du nombre de canaux qui reste à attribuer.

Le niveau de difficulté associé à une cellule est calculé à partir des quatres contraintes d'interférences et d'espacement. À partir de chacune d'elles nous évaluons le nombre de fréquences interdites. Observer que les contraintes co-canaux interdisent une seule fréquence, alors que les contraintes d'espacement et de canaux adjacents en interdisent deux, une de part et d'autre de la fréquence. De même chaque paire de fréquences distinctes génère 2 valeurs interdites et chaque triplet de fréquences distinctes en génère 3 pour les produits d'intermodulation du 3<sup>e</sup> ordre. Remarquez que dans le cas des canaux déjà affectés, seules les fréquences interdites positives sont comptabilisées.

Le choix des fréquences à attribuer se fait toujours en affectant à une cellule la première fréquence disponible, si cette affectation ne respecte pas toutes les contraintes, la prochaine valeur disponible sera considérée. Cette étape est répétée jusqu'à ce que nous obtenons une attribution valide ou jusqu'à l'épuisement l'ensemble  $F$  des fréquences disponibles. À ce moment, un retour arrière sera effectué et le processus se répète de nouveau. La recherche s'arrêtera quand une solution réalisable est obtenue où si une telle solution n'existe pas.

### Algorithme de recherche

Nous détaillons, ci-dessous, les principales étapes de l'algorithme de recherche tel qu'il a été défini dans la partie précédente.

- 1) Initialisation des structures de données associées aux problèmes.
- 2) **Tant** qu'on n'a pas affecté les fréquences à tous les canaux **faire**
- 3)     Choisir une cellule avec une demande non satisfaite.
- 4)     Choisir une fréquence disponible.
- 5)     **Tant** que l'affectation est invalide et qu'il reste des fréquences **faire**
- 6)         **Si** il y a suffisamment de fréquences pour satisfaire les demandes **alors**
- 7)             **Si** les contraintes sont toutes satisfaites **alors**
- 8)                 Attribuer la fréquence à la cellule.
- 9)                 Le canal est affecté.
- 10)             **Sinon**
- 11)                 Chercher la prochaine fréquence disponible.
- 12)             **Finsi ;**
- 13)     **Sinon**
- 14)     Retour en arrière.

- 15) Choisir une cellule non satisfaite.
- 16) Choisir une nouvelle fréquence disponible.
- 17) **Finsi ;**
- 18) **FinTant ;**
- 19) **Si** aucune fréquence est disponible pour le canal **alors**
- 20) Retour en arrière.
- 21) Choisir une cellule non satisfaite.
- 22) Choisir une nouvelle fréquence disponible.
- 23) **Finsi ;**
- 24) **FinTant ;**
- 25) **Si** toutes les fréquences sont attribuées **alors**
- 26) Imprimer la solution.
- 27) **Finsi ;**

### 3.6 Borne inférieure

Nous avons vu dans la section 3.4 que le problème d'affectation de fréquences est un problème relaxé des règles de Golomb. Cependant il demeure un problème combinatoire fort difficile à résoudre par méthode exacte. Une stratégie d'énumération implicite ne sera pas un choix judicieux. Néanmoins, une telle approche serait justifiable dans la mesure où nous désirons savoir si une solution satisfaisant les contraintes d'interférences et d'intermodulation existe.

Nous développons dans cette section, une borne inférieure pour réduire l'arbre de recherche. En se servant des contraintes d'espacement, nous pouvons calculer une fréquence minimale  $F(i)$  nécessaire pour satisfaire les demandes à l'intérieur d'une cellule  $C_i$ . Une première borne s'obtient en prenant le maximum des  $F(i)$  :

$$\forall i \quad \underline{s} = \max_i F(i).$$

Cette fonction ( $F(i)$ ) renvoie la valeur de la fréquence minimale nécessaire pour satisfaire toutes les demandes de la cellule. Elle est définie simplement par  $F(i) = esp_i * (N_i - 1)$ , où  $N_i$  est le nombre de fréquences demandées dans la cellule  $C_i$ . Dans la mesure où certaines fréquences sont déjà fixées et d'autres sont interdites par les contraintes, la borne inférieure s'obtient alors par :

$$\underline{s} = \max_i \left\{ \max\{1, f_{i, M_i - n_i - 1}\} + (n_i - 1) * esp_i \right\}$$

$M_i$  est le nombre de canaux demandés dans la cellule  $i$ .

$n_i$  est le nombre de canaux non affectés dans la cellule  $i$ .

$f_{i, M_i - n_i - 1}$  est la plus grande fréquence affectée à la cellule  $i$ .

Il s'agit bien sûr d'une borne inférieure assez rudimentaire, cependant nous pouvons améliorer légèrement sa valeur en considérant la fréquence minimale nécessaire pour satisfaire deux cellules qui sont fortement liées par les interférences et les

produits d'intermodulation. En évaluant la borne au fur et à mesure que les canaux sont fixés, nous pouvons améliorer davantage la fréquence minimale nécessaire. Nous espérons à partir de ces bornes restreindre suffisamment le domaine pour obtenir une réponse à savoir si une solution réalisable existe ou non pour le problème.

### 3.7 Résultats

Pour des fins d'expériences, nous avons extrait d'un problème réel un sous-problème comprenant 20 cellules avec une matrice d'intermodulation de densité maximale. Quatre fichiers de données sont fournis au programme, un premier indiquant les liens entre les sites et les cellules du sous-réseau, un deuxième, les demandes de chaque cellule, le troisième est une matrice comprenant les contraintes co-canaux, de canaux adjacents et d'espacements. Finalement le dernier fichier fournit la matrice d'intermodulation.

Nous avons combiné les matrices  $M^{co}$ ,  $M^{adj}$  et les contraintes d'espacement en une seule matrice notée  $M^{sep}$  afin de simplifier la lecture et réduire le nombre de fichiers à fournir au programme. Les éléments  $m_{ij}^{sep}$  de la matrice peuvent être interprétés comme l'ensemble des distances de séparation minimale entre les fréquences des cellules  $C_i$  et  $C_j$ . Les  $m_{ij}^{sep}$  auront les valeurs suivantes, soient 0, 1, 2 ou 17 qui sont déduites des contraintes d'inférences. Observer que les éléments de la diagonale auront tous la valeur 17 et les  $m_{ij}^{sep}$  pour tout  $i \neq j$  auront les valeurs déterminées par la règle suivante :

$$m_{ij}^{sep} = \begin{cases} 17 & \text{si } i = j, \\ 1 & \text{si } m_{ij}^{co} > m^{co}, m_{ij}^{adj} < m^{adj} \text{ et } i \neq j, \\ 2 & \text{si } m_{ij}^{adj} > m^{adj} \text{ et } i \neq j, \\ 0 & \text{autrement.} \end{cases}$$

Dans un premier temps, nous avons utilisé le programme pour résoudre un sous-problème extrait du réseau de la grande région de Montréal. La fréquence maximale disponible est fixée à 200 kHz. Une représentation graphique du réseau est illustrée à la figure 3.7. Chaque cellule est représentée par un pentagone. Les cellules  $C_i$  et  $C_j$  sont reliées par une droite indiquant que les produits de l'intermodulation doivent



être traités entre ces deux cellules. Les fichiers de données sont fournis à l'annexe A.

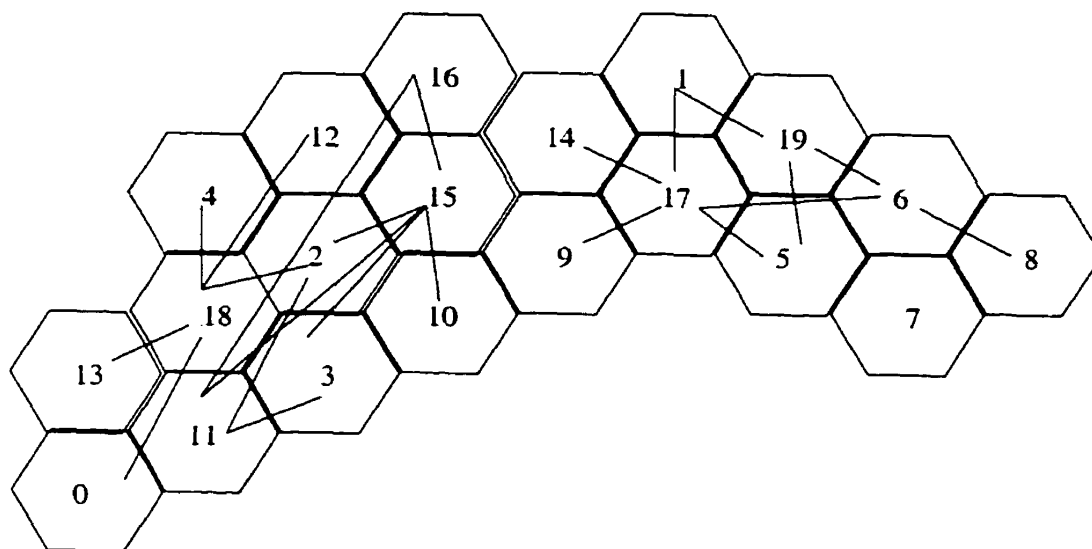


Figure 3.6 – Sous-réseau à 20 cellules

Une solution réalisable est obtenue et fournie dans le tableau 3.1. Observer que les produits de l'intermodulation sont concentrés dans deux régions du sous-réseau. Nous pouvons donc séparer ce dernier en deux sous-problèmes. Une coupure pourrait être effectuée entre les cellules 10, 15, 16 et 9 et 14. À partir de la région de gauche, nous avons ajouté 4 paires de cellules dans lesquelles nous devons traiter les produits de l'intermodulation. Ces couplets sont (2,3), (2,4), (2,12) et (2,16). L'ajout de ces derniers ont rendu le problème irréalisable. Nous pouvons constater à quel point les produits d'intermodulation sont difficiles à satisfaire dans le cadre des problèmes d'affectation de fréquences.

Tableau 3.1 – *Affectation de fréquences*

Cellule	fréquences
0	{13, 30, 47, 64, 81, 98}
1	{1, 18, 35, 52, 69, 86}
2	{6, 23, 40, 57, 74, 91}
3	{108, 125, 142, 159, 176, 193}
4	{6, 23, 40, 57, 74, 91}
5	{3, 20, 37, 54, 71, 88}
6	{2, 19, 36, 53, 70}
7	{4, 21, 38, 55, 72, 89}
8	{7, 24, 41, 58, 75, 92}
9	{5, 22, 39, 56, 73, 90}
10	{10, 27, 44, 61, 78, 95, 112, 129, 146}
11	{3, 20, 37, 54, 71, 88, 105, 122, 139, 156}
12	{14, 32, 100, 117, 134}
13	{12, 30, 47, 64, 81, 98}
14	{9, 26, 43, 60, 77, 94, 111}
15	{1, 18, 35, 52, 69, 86, 103}
16	{7, 21, 41, 58, 75, 92, 109, 126, 143}
17	{4, 21, 38, 55, 72, 89, 106, 123, 140, 157}
18	{1, 18, 35, 52, 69, 86, 103}
19	{8, 25, 42, 59, 76, 93, 110, 127}

### 3.8 Conclusion

Nous avons espéré utiliser les règles de Golomb dans la résolution des sous-problèmes d'affectation de fréquences. Cependant en examinant de plus près, nous constatons que ces règles n'étaient pas tout à fait équivalentes aux problèmes d'affectation de fréquences sous contraintes d'interférences et de produits d'intermodulation. Elles sont suffisamment différentes pour que nous ne puissions pas les utiliser.

Malgré qu'ils contiennent moins de contraintes que les problèmes de dérivation des règles de Golomb, les problèmes d'affectations de fréquences sous contraintes liées aux produits d'intermodulation demeurent extrêmement difficiles à résoudre de façon exacte. Le problème a tellement évolué au cours des années qu'on ne cherche plus à trouver une affectation optimale, mais tout simplement une affectation réalisable. Nous ne connaissons pas encore d'algorithmes efficaces pour ce type de problème, cependant il s'agit d'un domaine assez nouveau encore.

Bien qu'il ne soit pas pratique de résoudre ces problèmes par méthodes d'énumération implicite, il serait possible d'améliorer ces dernières en considérant de meilleures bornes qui tiendront compte des produits d'intermodulation. Compte tenu du niveau de difficulté des problèmes à résoudre et du fait que nous ne cherchons qu'une solution réalisable, les méthodes heuristiques conviendraient peut-être mieux à ces derniers.

# Bibliographie

- [1] ATKINSON, M. D., SANTORO, N., URRUTIA J. (1986). Integer Sets with Distinct Sums and Differences and Carrier Frequency Assignments for Nonlinear Repeaters. IEEE Transactions on Communications, volume 34, number 6, 614–617.
  
- [2] BABCOCK, W. C. (1953). Intermodulation interference in radio systems. Bell Syst. Tech. J, 31, 63–73.
  
- [3] BERGE, C. (1985). Graphs. Elsevier Science Publishers B.V., Amsterdam, volume 6, part 1. Dunod. 413 p.
  
- [4] BLOOM, G. S. and GOLOMB, S. W. (1978) Numbered Complete Graphs, Unusual Rulers, and Assorted Applications. Lecture Notes in Math., 642, Springer, Berlin, 53–65.
  
- [5] CHEN, Z., FAN, P. and JIN, F. (1992). Disjoint Difference Sets, Difference Triangle Sets, and Related Codes. IEEE Transactions on Information Theory, 38, 518–522.
  
- [6] CHEN, W. et KLØVE, T. (1991). Lower Bounds on Multiple Difference Sets. Discrete Mathematics, 98, 1, 9–21.
  
- [7] DEWDNEY A. K. (December 1985). The search for an invisible ruler that will help radio astronomers to measure the earth. Scientific American, 16–26.

- [8] DEWDNEY A. K. (March 1986). How a pair of dull-witted programs can look like geniuses on I.Q. tests. Scientific American, 14–21.
- [9] DOLLAS, A., McCRAKEN D., et RANKIN, W. T. (1998). A new Algorithm for Golomb ruler Derivation and Proof of the 19 Mark Ruler. IEEE Transactions on Information Theory, 44, 1, 379–382.
- [10] FANG, R. J. F. and SANDRIN, W. A. (1977). Carrier Frequency Assignment for Non-linear Repeaters. Comsat Tech. Rev., 7,1, 227–245.
- [11] Gardner, M. (March 1972). The graceful graphs of Solomon Golomb, or how to number a graph parsimoniously. Scientific American, volume 226, 108–112.
- [12] GIBBS, R. A. and SLATER, P. J. (1991). Distint distance sets in a graphe. Discrete Mathematics, 93, 155–165.
- [13] GOLOMB, S. W. (1972). How to Number a Graph. Graph Theory and Computing, New-York : Academic Press, R. Read, 23–37.
- [14] HACCOUN, D., LAVOIE P. et SAVARIA, Y. (April 1988). New Architectures for Fast Convolutional Encoders and Threshold Decoders. IEEE Journal on Select Areas in Communications, 6, 3, 547–557.

- [15] HACCOUN, D., LAVOIE P. et SAVARIA, Y. (February 1991) New VLSI Architectures for Fast Self-Decison Treshold Decoders. IEEE Transactions on Information Theory, 39, 2, 200-207.
- [16] HANSEN, P.B., JAUMARD, B. and MEYER, C. (March 1997). On Lower Bound for Numbered Complete Graphs.
- [17] KLEIBER, E. J. (1969). Some Difference Triangles for Constructing Self-Orthogonal Codes. IEEE Transactions on Information Theory, 16, 355-361.
- [18] KLØVE, T. (1988). Bounds on the Size of Optimal Difference Triangle Sets. IEEE Transactions on Information Theory, 34, 2, 355-361.
- [19] KLØVE, T. (1989). Bounds and Construction for Difference Triangle Sets. IEEE Transactions on Information Theory, 35, 4, 879-886.
- [20] KLØVE, T. (1990). Bounds and Constructions of Disjoint Sets of Distinct Difference Sets. IEEE Transactions on Information Theory, 36, 1, 184-190.
- [21] LEE, W. C.Y. (1995) Mobile Cellular Telecommunications, McGraw-Hill, inc., 463-533.
- [22] LINEBARGER, D. A., SUDBOROUGH I. H. and TOLLIS, I. G. (1993). Difference Bases and Sparse Sensor Arrays. IEEE Transactions on Information Theory, 39, 716-721.

- [23] LORENTZEN, R. and NILSEN R. (1991). Application of Linear Programming to the Optimal Difference Triangle Set Problem. IEEE Transactions on Information Theory, 37, 5, 1486–1488.
- [24] RANKIN, W. T. (December 1993). Optimal Golomb Rulers : An Exhaustive Parallel Search Implementation. Master's Thesis, Duke University, Durham, NC.
- [25] ROBINSON, J. P. and BERSTEIN, A. J. (January 1967). A Class of Binary Recurrent Codes with Limited Error Propagation. IEEE Transactions on Computers, 13, 1, 106–113.
- [26] SHEARER, J. B. (1990) Some New Optimum Golomb Rulers. IEEE Transactions on Information Theory, 36, 1, 183–184.
- [27] DOLIDAY, S. W., HOMAIFAR, A. and LEBBY, G. L. (1995) Genetic Algorithm Approach to the Search for Golomb Rulers. Proceedings of the Sixth International Conference on Genetic Algorithms, University of Pittsburgh PA, 1, 528–535.
- [28] WU, W. W. (September 1975). New Convolutional Codes-Part 1. IEEE Transactions on Information Theory, 23, 9, 942–955.
- [29] WU, W. W. (January 1976). New Convolutional Codes-Part 2. IEEE Transactions on Information Theory, 24, 1, 19–33.

- [30] WU, W. W. (September 1976). New Convolutional Codes-Part 3. IEEE Transactions on Information Theory, 24, 9, 946-955.
- [31] RANKIN, W. T. et GARRY, M. In search of the optimal 20, 21 & 22 Marks Golomb rulers. [http ://members.aol.com/golomb2](http://members.aol.com/golomb2)
- [32] RANKIN, W. T. et GARRY, M. Getting Started : Download The Files You Need. [http ://members.aol.com/golomb20/download.htm](http://members.aol.com/golomb20/download.htm)
- [33] SHEARER, J. [http ://www.research.ibm.com/people/s/shearer/grtab.html](http://www.research.ibm.com/people/s/shearer/grtab.html)
- [34] SHEARER, J. Welcome to James B. Shearer's home page.  
[http ://www.research.ibm.com/people/s/shearer](http://www.research.ibm.com/people/s/shearer)
- [35] MILLER, L. [http ://http ://www.cuug.ab.ca/ millerl/g3-records.html](http://http://www.cuug.ab.ca/millerl/g3-records.html)
- [36] COOLSAET, K. [http ://gonzo.hogent.be/~kc/cds/welcome.htm](http://gonzo.hogent.be/~kc/cds/welcome.htm)
- [37] COOLSAET, K. [http ://gonzo.hogent.be/~kc/cds/semiaff.cc](http://gonzo.hogent.be/~kc/cds/semiaff.cc)
- [38] COOLSAET, K. [http ://gonzo.hogent.be/~kc/cds/part7.htm](http://gonzo.hogent.be/~kc/cds/part7.htm)
- [39] COOLSAET, K. [http ://gonzo.hogent.be/~kc/cds/part10.htm](http://gonzo.hogent.be/~kc/cds/part10.htm)
- [40] GARRY, M. OGR-22 Search Speeds  
[http ://members.aol.com/golomb20/timing2.htm](http://members.aol.com/golomb20/timing2.htm)



[41] HEWGILL, G. <http://www.hewgill.com/ogr/>

## Annexe A

### Le problème d'affectation

## A.1 Les cellules d'un site

c000 0

c001 1

c002 2

c003 3

c004 4

c005 5

c006 5

c007 6

c008 6

c009 7

c010 8

c011 8

c012 9

c013 10

c014 11

c015 11

c016 12

c017 13

c018 14

c019 15

## A.2 Demande en canaux de chaque cellule

c0000 6  
c0001 6  
c0002 6  
c0003 6  
c0004 6  
c0005 6  
c0006 5  
c0007 6  
c0008 6  
c0009 6  
c0010 9  
c0011 10  
c0012 5  
c0013 6  
c0014 7  
c0015 7  
c0016 9  
c0017 10  
c0018 7  
c0019 8

### A.3 Matrice $M^{sep}$

c0000	17 2 0 0 1 1 1 1 2 2 0 0 0 0 0 0 0 0 0 0
c0001	2 17 0 0 1 1 1 1 2 2 0 0 0 0 0 0 0 0 0 0
c0002	0 0 17 2 0 0 0 0 0 0 2 2 1 1 0 0 1 1 1 1
c0003	0 0 2 17 0 0 0 0 0 0 2 2 1 1 0 0 1 1 1 1
c0004	1 1 0 0 17 2 1 1 1 1 0 0 0 0 0 0 0 0 1 1
c0005	1 1 0 0 2 17 1 1 1 1 0 0 0 0 0 0 0 0 1 1
c0006	1 1 0 0 1 1 17 2 0 0 1 1 0 0 0 0 0 0 1 1
c0007	1 1 0 0 1 1 2 17 0 0 1 1 0 0 0 0 0 0 1 1
c0008	2 2 0 0 1 1 0 0 17 2 0 0 0 0 0 0 0 0 0 0
c0009	2 2 0 0 1 1 0 0 2 17 0 0 0 0 0 0 0 0 0 0
c0010	0 0 2 2 0 0 1 1 0 0 17 2 1 1 1 1 1 1 2 2
c0011	0 0 2 2 0 0 1 1 0 0 2 17 1 1 1 1 1 1 2 2
c0012	0 0 1 1 0 0 0 0 0 0 1 1 17 2 2 2 2 2 1 1
c0013	0 0 1 1 0 0 0 0 0 0 1 1 2 17 2 2 2 2 1 1
c0014	0 0 0 0 0 0 0 0 0 0 1 1 2 2 17 2 2 2 0 0
c0015	0 0 0 0 0 0 0 0 0 0 1 1 2 2 2 17 2 2 0 0
c0016	0 0 1 1 0 0 0 0 0 0 1 1 2 2 2 2 17 2 0 0
c0017	0 0 1 1 0 0 0 0 0 0 1 1 2 2 2 2 2 17 0 0
c0018	0 0 1 1 1 1 1 1 0 0 2 2 1 1 0 0 0 0 17 2
c0019	0 0 1 1 1 1 1 1 0 0 2 2 1 1 0 0 0 0 2 17

#### A.4 Matrice $M^{mod}$

```

00000000000000000000000010
000000000000000000000000101
0000000000000010000100010
0000000000000010000100000
00000000000000000000000010
000000000000000000000000101
0000000011000000000000001
0000000100000000000000000
0000000100000000000000000
000000000000000000000000100
00000000000000000000000010000
0011000000000000000011000
00000000000000000000000010
00000000000000000000000010
000000000000000000000000100
00110000000011000001000
00000000000001000010000
010001000100000100000
10101000000000110000000
01000110000000000000000

```

# **Annexe B**

## **Les contributions**

## **B.1 Les contributions**

### **B.1.1 Les règles de Golomb**

1. Amélioration des tests de réduction.
2. Développer les formules analytiques pour le calcul des bornes inférieures.
3. Amélioration de l'algorithme original.
4. Implantation d'une deuxième stratégie de recherche.

### **B.1.2 Le problème d'affectation de fréquences**

1. Montrer que le problème d'affectation de fréquences sous contraintes des produits de l'intermodulation sont différentes du problème des règles de Golomb.
2. Développer une borne inférieure.